(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 December 2002 (12.12.2002)

PCT

(10) International Publication Number
WO 02/099683 A1

(51) International Patent Classification[7]: G06F 15/177

(21) International Application Number: PCT/US02/09179

(22) International Filing Date: 27 March 2002 (27.03.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/279,059 27 March 2001 (27.03.2001) US
60/311,268 9 August 2001 (09.08.2001) US

(71) Applicant: **NETBOTZ, INC.** [US/US]; 11044 Research Blvd., Suite C-100, Austin, TX 78759 (US).

(72) Inventors: **CHILDERS, Sloan, A.**; 1402 Beth Lane, Round Rock, TX 78664 (US). **ELDERTON, John**; 3112 Burks Lane, Austin, TX 78732 (US). **PRIMM, Michael**; 10237 Matoca Way, Austin, TX 78726 (US).

(74) Agent: **HULSEY, William, N.**; Hughes & Luce, L.L.P., Suite 2800, 1717 Main Street, Dallas, TX 75210 (US).

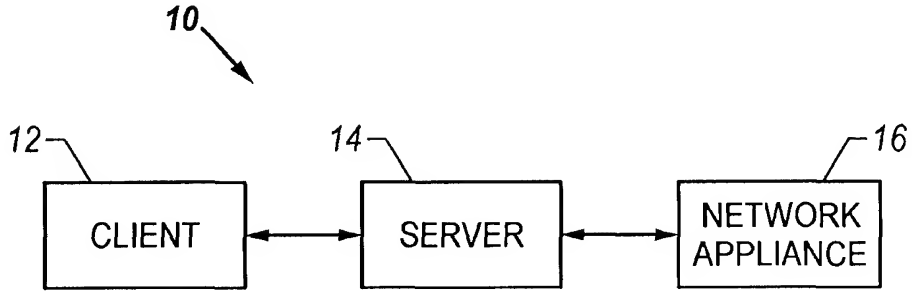
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHODS FOR DISPLAYING PHYSICAL NETWORK TOPOLOGY AND ENVIRONMENTAL STATUS BY LOCATION, ORGANIZATION, OR RESPONSIBLE PARTY



(57) Abstract: The invention is directed to a system (10) for remote monitoring of a space and equipment. The system has network appliances (16), a server (14), and a client (12). The server (14) receives data from a network appliance (16). The server (14) may then deliver an application and data to the client (12) for viewing and manipulating the data. The client (12) may display the data as a mapping, displaying icons associated with the network appliances (16). The client (12) may manipulate the organization of the data, the configuration settings of the network appliances (16) and store map and graph configurations.

WO 02/099683 A1

METHODS FOR DISPLAYING PHYSICAL NETWORK TOPOLOGY AND ENVIRONMENTAL STATUS BY LOCATION, ORGANIZATION, OR RESPONSIBLE PARTY

TECHNICAL FIELD OF THE INVENTION

This invention relates in general to a system and method for monitoring network equipment. More specifically, the invention relates to a system and method of monitoring network-enabled sensor equipment from a remote location.

RELATED APPLICATIONS

This application is a continuation-in-part of U.S. patent Application, Serial No. 09/429,504, filed October 27, 1999 entitled: "METHOD AND SYSTEM FOR MONITORING COMPUTER NETWORKS AND EQUIPMENT", and is incorporated herein by reference in its entirety.

This application is a continuation-in-part of U.S. patent Application, Serial No. 10/057,563, filed January 25, 2002 entitled: "METHOD AND SYSTEM FOR A SET OF NETWORK APPLIANCES WHICH CAN BE CONNECTED TO PROVIDE ENHANCED COLLABORATION, SCALABILITY, AND RELIABILITY", which claims priority of U.S. provisional Application No. 60/264,445, filed January 26, 2001 entitled: "METHOD AND SYSTEM FOR A SET OF NETWORK APPLIANCES WHICH CAN BE CONNECTED TO PROVIDE ENHANCED COLLABORATION, SCALABILITY, AND RELIABILITY and is incorporated herein by reference in its entirety.

This application claims priority of U.S. provisional Application, No. 60/279,059, filed March 27, 2001 entitled: "SENSOR PLAYBACK SYSTEM AND METHOD", and is incorporated herein by reference in its entirety.

This application claims priority of U.S. provisional Application, Serial No. 60/311,268, filed August 9, 2001 entitled: "METHODS FOR DISPLAYING PHYSICAL NETWORK TOPOLOGY AND ENVIRONMENTAL STATUS BY LOCATION, ORGANIZATION, OR RESPONSIBLE PARTY", and is incorporated herein by reference in its entirety.

BACKGROUND OF THE INVENTION:

Data traffic on networks, particularly on the Internet, has increased dramatically over the past several years, and this trend will continue with the rapid growth of e-commerce and other services on the Internet requiring greater bandwidth. With this increase in data traffic on networks, there has been a corresponding increase in the number of computer equipment rooms, known as "server rooms," used to house the equipment necessary to support data traffic routing. Furthermore, the increasing dependency of companies on their Internet presence has created an urgency to keep the server rooms up and running at all times. Industry estimates show that there are over 400,000 such rooms currently in existence in the United States.

The growth in Internet traffic has prompted many businesses to construct a server room to allow their employees to access Internet information or enable e-commerce and store data. As such, continuous server up time has become important. Keeping track of numerous computers, along with associated bridges, routers, backup power supplies, etc., can be a formidable task. A large company with server rooms in more than one city might well be faced with spending thousands of dollars on software packages to keep their equipment running. Dedicated technicians are also needed to monitor network equipment and issue work orders to repair failed units.

While reliable, modern computer systems cannot tolerate excess heat, dust or humidity. Heat can rapidly cause equipment deterioration. Failure of cooling fans can reduce equipment lifetime to days or hours. A single high-speed LAN (local area network) failure can cause slow

system response. These and other such failures within the equipment in a server room occur routinely and can cause great disruption to a business.

Typical solutions only permit inspection of devices on a local basis. Others permit a technician to inspect geographically diverse installations from a central console. However, all of these solutions are expensive to implement, complex, and difficult to maintain and train personnel to use them.

As a result, small to medium companies having small to medium networks are left in the position of requiring a means to monitor and maintain their computer network equipment from failing while not having the resources to afford the high-priced solutions. Many firms cannot afford a high-end solution or simply do not have the time and resources to train their IT personnel to learn and use complex systems. Instead, the typical monitoring method in many such companies is user complaints to the IT manager to indicate when a problem has occurred.

This is especially true for companies having multiple server rooms and that have concerns about routine access to each of these rooms. Additionally, concerns exist with current solutions regarding the manpower intensiveness of these solutions. Most network monitoring solutions can consume a full or part-time employee. The financial justification for these systems is, therefore, difficult because network equipment typically fails yearly or on a disaster basis, and the cost of recovery is seen as less than that of maintaining a full-time employee to routinely monitor the equipment.

Similar concerns exist for monitoring rack-mounted components. Typical problems include localized environmental excesses leading to failure. Another problem is theft. Typical monitoring solutions do not provide for video imaging of remote server locations over a network. Computer equipment is typically placed in server rooms for two reasons: security and

environmental control. Remote video imaging of a server room over a network can provide for maintaining security of the equipment despite the lack of a physical presence on site.

A typical computer room can house hundreds of devices, ranging from expensive server grade computers to bridges, routers, uninterruptible power supplies and telephone equipment. A server room's environment requires monitoring because out-of-limit environmental variables can eventually affect the equipment in the room. For example, high temperatures, humidity (for example, from water leaks), or lack of airflow can detrimentally affect the equipment. Similarly, alarms, such as smoke and fire alarms, or the status of room openings, are important to determine. While the expense of replacing server room components if they fail is great, currently existing monitoring solutions are not cost effective for smaller-sized companies to implement despite the potential costs of such losses.

These typical monitoring systems use a centralized application. While these mechanisms can be quite effective, they introduce additional costs, through additional software, hardware, configuration, administration, and network bandwidth.

Beyond the application to server rooms and rack mountings of network equipment, various other monitoring systems suffer from the same failures and deficiencies associated with information accessibility, organization, and presentation.

As such, many typical network monitoring systems suffer from deficiencies in information accessibility, organization, and presentation. Many other problems and disadvantages of the prior art will become apparent to one skilled in the art after comparing such prior art with the present invention as described herein.

SUMMARY OF THE INVENTION:

Aspects of the invention may be found in a system for remote monitoring of network appliances. The system may have a server. The server may be in communication with a network appliance. Further, the server may be in communication with the client machine. The server may function to download sensory data from the network appliance or appliances and store the information. Further, the server may function to upload configuration data to the network appliance or appliances.

In addition, the server may communicate with the client machine. The server may transfer software to the client machine. The software may permit the client machine to access sensory data on the server and to manipulate configuration data.

The server, network appliance, and client may communicate through an interconnected network. The interconnected network may be a global network, wireless network, wide area network, and local area network, among others.

Aspects of the invention may also be found in a server. The server may be in communication with the network appliances. Further, the server may be in communication with the client machine. A server may download information from the network appliances associated with sensor data. Further, the server may upload configuration data to the network appliances. A server may also supply software to the client machine. The software may enable the client machine to access data on the server, manipulate the data, display the data and change configuration data associated with the network appliances.

The server may also have map configuration data. These data may be transferred to the client machine and used to display data associated with the network appliances. Further, the data may be displayed as icons arranged on a display. These icons may be organized in a manner that represents physical location, status, and function, among others. Furthermore, these

icons may be superimposed on a graphic element. The graphic element may be a map, an image, or a plot, among others.

The server may also function to store image data associated with the network appliances. The image data may be stored in a manner that associates the image data with other sensory data or sensory events occurring on the network appliances. The image data may be a still image, an infrared image or a movie, among others.

Aspects of the invention may also be found in a client machine and software operable to run on the client machine. The software may be acquired from the server. The software may enable the client machine to access information associated with the network appliances. This information may be sensory data and/or configuration data, among others. The software may also enable the client machine to display the data. For example, the data may be displayed in the form of a map, a graph, or a table, among others. The software may also enable the client machine to manipulate data, map configuration data, and configurations associated with network appliances, among others. For example, the client may customize a map by specifying the organization of icons associated with network appliances and their data associated therewith. Alternately, the client may manipulate the access of other users to the map. The client may also manipulate configurations associated with network appliances by changing a parameter associated with several network appliances to a same value for each of the several network appliances. The client may also manipulate a graph, such that data associated with multiple sensors from multiple network appliances may be displayed. The client may also display image data associated with sensors and/or physical events associated with the network appliances.

Aspects of the invention may also be found in a map configuration data. The map configuration data may be automatically generated or configured by a client, among others. Further, the map configuration data may have restricted permissions such that the map may be

viewable by one or few other users. Further, the map may have a configuration such that users are given varying permissions associated with viewing and changing the map. The map may also display icons super imposed over a background. For example, the background may depict a map of a physical location over which icons are displayed representing the physical location of an associated network appliance.

Further aspects of the invention may be found in a method for mass configuration of network appliances. The method may include changing a parameter associated with several network appliances to a same value for each network appliance. A client machine may perform the change. The change may then be stored on a server. A network appliance may then ping the server. The server may respond with new configuration data. Further, the network appliance may respond with a confirmation and the server may respond to the confirmation with additional data, if available.

Another aspect of the invention may be found in a graph. The graph may be a graph of data associated with network appliances. The graph may be displayed on client machine. Further, the graph may be associated with various sensors associated with multiple appliances. The sensors may be of a same type, a varying type, or a combination of types, among others.

Further aspects of the invention may be found in an image. The image may be acquired from a image acquisition enabled network appliance. The network appliance may acquire the image in response to an event on another network appliance, or the same network appliance, or combinations of network appliances, among others. The image may be transferred to a server for storage. A server may store the image in a manner such that the image is associated with the event. Further, the image may be stored such that it is associated with a time the image was taken, the time of the event, or other factors. Further, the image may be associated with various network appliances.

As such, a system for remote monitoring of network equipment is described. Other aspects, advantages and novel features of the present invention will become apparent from the detailed description of the invention when considered in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS:

For a more complete understanding of the present invention and advantages thereof, reference is now made to the following description taken in conjunction with the accompanying drawings in which like reference numbers indicate like features and wherein:

Figure 1 is a schematic block diagram of a system, according to the invention;

Figure 2A is a schematic block diagram of an exemplary embodiment of the system as seen in Figure 1;

Figure 2B is another schematic block diagram of another exemplary embodiment of the system as seen in Figure 1;

Figure 2C is a schematic block diagram of a further exemplary embodiment of the system as seen in Figure 1;

Figure 3A is a block flow diagram of an exemplary method for use by the system as seen in Figure1;

Figure 3B is a block flow diagram of an exemplary method for use by the system as seen in Figure 1;

Figure 4 is a block diagram of an exemplary embodiment of the client as seen in Figure1;

Figure 5 is a block diagram of an exemplary embodiment of the server as seen in Figure 1;

Figure 6 is a block diagram of an exemplary embodiment of a network appliance as seen in Figure 1;

Figure 7 is a schematic block diagram of an exemplary embodiment of a map as seen in Figure 5;

Figure 8A is schematic block diagram of an exemplary embodiment of the map as seen in Figure 7;

Figure 8B is a schematic block diagram of another exemplary embodiment of the map as seen in Figure 7;

Figure 8C is a schematic block diagram of a further exemplary embodiment of the map as seen in Figure 7;

Figure 8D is a schematic block diagram of another exemplary embodiment of the map as seen in Figure 7;

Figure 9A is a block diagram of an exemplary embodiment of a configuration of several network appliances as seen in Figure 5;

Figure 9B is a block diagram of another exemplary embodiment of a configuration as seen in Figure 5;

Figure 10 is a block flow diagram of an exemplary method for use by the system as seen in Figure 1;

Figure 11 is a block flow diagram of an exemplary method for use by the system as seen in Figure 1;

Figure 12 is a schematic diagram of a exemplary embodiment of a grouping according to Figure 5;

Figure 13A is a block diagram of an exemplary embodiment of a display for use by the system of Figure 4;

Figure 13B is a block diagram of an exemplary embodiment of a display for use by the system as seen in Figure 4;

Figure 13 C is a block diagram of a further exemplary embodiment of a display for use by the system as seen in Figure 4;

Figure 14 is a chart of an exemplary embodiment as displayed by the system of Figure 4; and

Figure 15 is a schematic block diagram of an exemplary embodiment of a display for use by the system of Figure 4.

Corresponding reference numerals indicate corresponding parts throughout the several views of the drawings.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT:

Figure 1 is a schematic block diagram of the system according to the invention. The system 10 has a server, a client 12 and a network appliance 16. The server 14 is connected to one or more network appliances 16 through an interconnected network. The server 14 may function to transfer sensor data from the network appliance 16 and transfer configuration data to the network appliances 16. The server 14 is also connected to a client machine 12. The client machine 12 may access, display and/or manipulate data stored on the server 14. In this manner, the client 12 may remotely monitor network appliances 16 and the client 12 may reconfigure the network appliances 16.

The client 12 may be connected to the server 14 through an interconnected network. Further, the server 14 may be connected to the network appliance 16 through an interconnected

network. The interconnected network may take various forms. These forms may include a global network, wide area network, local area network, wireless network, phone systems, and satellite communications systems, among others. Further, these networks and systems may use various method, protocols, and standards, including, among others, ethernet, wireless ethernet, TCP/IP, HTTP, FTP, SNMP, Blue Tooth, and others. In addition, various security methods may be used in transferring data, including SSL, among others. Further, a user-controlled level of security may be provided. A standard protocol may allow the client and server to be physically located on separate sides of a firewall, adding another level of security to the customer.

In addition, the client 12 may acquire instructions for accessing, displaying and manipulating data from the server 14. These instructions may also be transferred by the server from the server 14 on an as needed basis.

In one exemplary embodiment, the server 14 may communicate with one or more network appliances 16. The one or more network appliances 16 may be located in a server room. The one or more network appliances 16 may have sensors for sensing environmental conditions and security states of the server room.

For example, the network appliances 16 may collect data associated with temperature, humidity, door sensors, alarms, power quality, motion detectors and cameras, among others. The network appliances 16 may, for example, communicate with the server 14 through hypertext transfer protocols. In one exemplary embodiment, the network appliances 16 are connected to an interconnected network, such as a local area network, wide area network, global network, and wireless network, among others. The network may, for example, use a TCP/IP protocol communications method. The network appliances 16 may, for example, communicate with the server 14 using a hypertext transfer protocol.

For example, the network appliances 16 may ping a server 14 with an HTTP method communication. The server 14 may respond to that HTTP ping method communication with data associated with the configuration of the network appliance 16. Alternately, the network appliances 16 may use the HTTP method communication to transfer data to the server 14. In one embodiment, the network appliance 16 may use an HTTP Post method to send information relating to alarms and alerts. Some alarms and/or alerts may have associated image data which may be stored on the server 14. Furthermore, the server may associate the image data with the alert. Alerts delivered via HTTP Posts may allow other appliances to communicate and deliver information to servers that cannot initiate communications with the Appliances, for example, due to firewalls or intermittent network connectivity. This approach may provide superior reliability, security, and connectivity to conventional SNMP alert delivery.

The HTTP Post method may also be used to implement periodic posting of data from the network appliance to the server. The end-user may also configure appliances to periodically deliver their sensor data to the present invention, "pushing" the data to the server instead of having the server "pull" the data from the appliance. This mechanism allows the server to collect and record data from appliances that it is not capable of initiating communications with, such as appliances located behind a fully blocking firewall to inbound network requests. The delivery of this data may be set to require a user-id and password, allowing the present invention to authenticate the delivered data. The same transactions used for communicating the current sensor values and states may be used to verify status. If the delivery of the data is significantly overdue (i.e. by some period of time, or some number of scheduled Posts are missing), the Server will declare the Appliance "offline" or "missing in action".

In another embodiment, the server 14 may communicate with the network appliance 16 using an HTTP Get call. However, the server 14 and network appliances 16 may use various

communications methods. These methods may include file transfer protocol, hypertext transfer protocol, SNMP, among others. Further, the communications may include messages associated with HTML, XML, HTTP post, HTTP get, compressed data, and image data, among others. The communication may occur on intervals. These intervals may be fixed periodically, vary with date or time, be adjustable, or any combination, among others. In addition, timeouts and retries may be configured.

Further, the server may attempt to find network appliances through discovery. For example, the server may attempt to communicate with each possible address in a given IP address range. In addition, it may attempt to communicate with each of a specified set of ports that the user has configured the HTTP servers on their appliances to use.

The ability to schedule a discovery or collect environmental sensor data during a control window makes life easier for network administrators to reduce network management traffic during peak hours. This approach may allow the user to configure which days of the week to scan for their appliances, as well as what time of day to do the scan. This feature may also allows the user to find appliances located at network sites that are only "dialed up" during certain scheduled times of days, without wasting time and effort attempting to discover them when they are not connected to the central site.

The present invention supports an arbitrary number of discovery policies, allowing discovery to be fine-tuned for multiple sites and different customer policies.

The system may also support "discovering" appliances by handling Appliance-initiated HTTP Posts. When an Appliance issues a Post to the Server, the server will determine if the Appliance is one already managed by the Server. If not, the Appliance will be automatically added, either unconditionally or if it meets certain criteria configured by the user (i.e. only devices on certain subnets, certain models, or matching membership criteria for certain Groups

(see 3.9)). The Server's response to the Post may be used to tell the Appliance how often to check-in in the future (if it is accepted) or to not Post again in the future (if it is rejected), among others.

The server 14 may communicate with a client machine 12. For example, the client machine 12 and server 14 may be coupled to an interconnected network. The interconnected network may take various forms. These forms may include global networks, local area networks, wide area networks, wireless networks, phone switching networks, and others. Further, these networks may use various protocols, such as TCP/IP.

In one exemplary embodiment, the client machine 12 may communicate with the server 14 using hypertext transfer protocols. For example, the client machine 12 may have a web browser that communicates with the server 14. The web browser may be a JAVA enabled browser. For example, a JAVA enabled browser may download an applet from the server 14. The applets may enable the client machine to access, display, and/or manipulate data stored on the server 14. For example, the client machine 12 may be able to access information associated with sensor data, configuration data, image data, network appliance status, and map configuration data, among others. In one exemplary embodiment, the client machine 12 may query the server using SQL to retrieve the desired data. However, various other methods may be used to retrieve data.

The client machine 12 may then display the data in various formats including tables, maps, and graphs, among others. Furthermore, the client 12 may, in one exemplary embodiment, dynamically load JAVA programming object classes for viewing, accessing, and/or manipulating various data. Most of the HTTP replies returned from the server are in plain ASCII text. However there are several situations where binary transfers of Java Objects are far more efficient. For these scenarios, a Network Class Loader may be implemented so the

server can create complex return-objects for the client. Since the client may be relatively small, a mechanism may provide the underlying Object code to the client before it receives the Object itself. The Network Class Loader is that solution. In other words, the client can make a request to the server and receive both an Object containing data, and the code necessary to decode and execute the returned Object within the client's application environment.

This feature may further enhance the ability of third-party developers (both end-user and ISVs) to extend the present invention, since the definitions of these interfaces and the classes returned can be published without requiring the ISV to include potentially obsolete versions of the class implementations in their delivered code (since the up-to-date versions will be served to the application from the present invention using the Network Class Loader). For compression purposes, returned objects from the server may utilize the Object serialization standard put forth by Sun Microsystems in the Java Runtime Environment.

The client machine 12 may also manipulate and organize data. In one exemplary embodiment, the client machine 12 may establish dynamic groups, organized by chain of command, business infrastructure, or physical location, among others. These groups may be displayed in a tree structure. Further, these groupings may, for example, be implemented using dynamically created queries.

However, the client machine may have various embodiments. Furthermore, the client machine may communicate with the server 14 through various protocols. These protocols may include FTP, HTTP, SNMP, among others. In an alternate embodiment, the client machine 12 may contain software. The software may be functional to acquire and load various programming objects and classes. The software may also be written in various languages such as JAVA, C++, Visual Basic, among others.

The server 14 may also communicate to the client machine 12 an alert associated with storage capacity. Further, the server 14 may implement automated backup.

Figure 2A is a schematic block diagram of an exemplary embodiment of the system as seen in Figure 1. The system 30 may have a server 34 connected to an interconnected network 32. In addition, the system 30 may have client machines 36, 38, network appliances 40, 42, or third party appliances 44 connected to a network 32, among others. The server 34 may function to store information associated with the network appliances. This information may include sensor data, configuration data, image data and map configuration files, among others. The data or information may be down loaded by the server 34 from the network appliances 40, 42. Alternately, the network appliances 40, 42 may transfer data or information to the server 34 through the interconnected network 32.

Furthermore, the server may acquire data from a third party appliance 44 through the interconnected network 32. A server 34 may store, group and organize the information and data. Further, the server may supply the information to one or more client machines 36, 38, through the interconnected network 32.

One or more client machines 36, 38, may communicate with the server 34 through an interconnected network 32. The clients 36, 38 may access data, display, and manipulate data, among others. Furthermore, the clients 36, 38 may acquire instructions and/or programs associated with accessing the data from the server 34.

However, the server 34, the network appliances 40, 42, the third party appliance 44 and the clients 36, 38 may or may not be connected to the same interconnected network. Moreover, these elements may be configured separately, together, or in various combinations, among others.

For example, Figure 2B is a schematic block diagram of an exemplary embodiment of the system as seen in Figure 1. The system has a server connected to two interconnected networks 52, 54. The interconnected network 52 also connects to client machines 58, 60, and 62. The interconnected network 54 may connect to one or more network appliances 64, 66, 68, and/or third party appliances 69. A server 56 may transfer information to and from the one or more appliances 64, 66, 68 and/or the third party appliances 69 through the interconnected network 54. This information may be sensor data, configuration data, and images, among others.

The server 56 may store the information and supply that information to client machines 58, 60, 62. The client machines 58, 60, 62 may, for example, access, display and/or manipulate the data associated with the network appliances 64, 66, 68 and third party appliances 69. Further, the client machines 58, 60, 62 may acquire from the server 56, instructions, objects, classes, and programs, among others, for accessing, displaying and manipulating the data associated with the network appliances 64, 66, 68 and third party appliances 69, as stored on the server 56.

Further, Figure 2C is a schematic block diagram of a further exemplary embodiment of the system as seen in Figure 1. The system 70 has a server 76. The server 76 may be connected to a network appliance A 84 or optionally connected to a network appliance B 88. Network appliance A 84 and network appliance B 88 may be connected to an interconnected network 74. In addition, network appliance 86 and a third party appliance 89 may be connected to the interconnected network 74. The server 76 may be connected to the network appliance A84 through various means. These means may include a global network, wide area network, local area network, wireless network, phone systems, and satellite communications systems, among others. Further, these networks and systems may use various method, protocols, and standards,

including, among others, ethernet, wireless ethernet, TCP/IP, HTTP, FTP, SNMP, Blue Tooth, and others.

In addition, the server 76 may be connected to network appliance B 88 through various means. These means may include a global network, wide area network, local area network, wireless network, phone systems, and satellite communications systems, among others. Further, these networks and systems may use various method, protocols, and standards, including, among others, ethernet, wireless ethernet, TCP/IP, HTTP, FTP, SNMP, Blue Tooth, and others.

Moreover, the server 76 may be connected to network appliance A 84 and network appliance B 88 through the same, different, or various combinations, among others, of interconnected communication methods.

In addition, the server 76 may be connected to one or more client machines 78, 80 82 through an interconnected network 72. The client machines 78, 80, 82, may, through the interconnected network 72, access, display, and manipulate data associated with the network appliances 84, 86, 88 and/or third party appliances 89 as stored on the server 76. Furthermore, the client machines 78, 80, 82 may acquire from the server 76, instructions, objects, and classes, among others, for accessing, displaying and manipulating data as stored on the server 76.

The server 76 may store data associated with the network appliances 84, 86, 88 and third party appliances 89. This information may include sensor data, configuration data, map configuration data, groupings and associations, accessibility information, and image data, among others. The server, may, for example, communicate with network appliance A 84 to transfer the data. Alternately, the server 76 may communicate with network appliance B 88 to transfer the data. In one exemplary embodiment, network appliance A 84 may act as an intermediate between network appliances 86, 88, third party appliances 89 and the server 76. Network

appliance A 84 may function as an intermediary by storing a directory of data, acting as a proxy, or acting as a data reciprocal, among others.

However, the elements as seen in Figures 2A, 2B and 2C may configured in various combinations, together or separate, among others. As such, various configurations may be envisaged.

The client machine may connect to a server to acquire data. The data may change or fluctuate dynamically on the varying conditions at a network appliance. Various methods may be used to update the data as displayed on the client machine. For example, the client machine may stay connected to the server continuously. However, continuous connection to the server may represent a burden to the network. Alternately, the client machine may periodically download updates from the server. However, the data integrity as displayed on the client machine may suffer, as changes to the data on the server may not be seen on the client machine.

In another embodiment, combinations of these methods may be used. For example, Figure 3A shows a method for communicating between the client and server. In this exemplary method 90, the client may connect to the server as seen in a block 92. The client may wait and acquire data updates as they occur as seen in a block 94. The client may then disconnect as seen in a block 96. During this time, the data displayed may not be updated by the server. The client may then wait for a period of time as seen by a block 98 and reconnect. In this manner, while connected, the data integrity will be maintained. However, periodic disconnection will improve network communications or decrease network load. The periods of connecting and disconnecting may be constant, manipulated, or fluctuate with activity or demand. For example, during periods o f low activity, the disconnect period may be increased. Alternately, during periods of high activity, the connect period may be increased. Furthermore, these periods may

be changed in response to user interaction, user activity, and network appliance activity, server activity, among others.

The client may connect using various methods and protocols, among others. Alternately, the client may use a ping method or various GET or POST methods to contact the server at varying intervals. These intervals may be increased or reduced to simulate a connection or wait period.

In response, the server may alter its approach to collecting and transferring data. For example, Figure 3B shows a block flow diagram of an exemplary method for use by the server. The method 100 involves cueing data as seen in a block 102 when the client machine is disconnected. Then, data is automatically delivered, as seen in a block 104, when the client machine is connected. However, various other embodiments may be envisaged for communicating between the client and the server.

Figure 4 shows an exemplary embodiment of a client machine as seen in Figure 1. The client machine 110 may have a processor 112, programmable circuitry 114, one or more network interfaces 116, one or more user interfaces 118, and storage mediums 120, among others. A storage mediums 120 may store application data. Further the storage mediums may store downloaded data and information 128. However, the client 110 may have various configurations. These elements may or may not be included. Further, these elements may be separate, together, or in various combinations, among others.

The processor 112 may function to interpret the instructions and application data. The processor may take various forms. These forms may include CPUs, embedded processors, JAVA enabled processors, and various computational circuitry, among others. Further, the processor may operate with an operating system such as Windows 95, Windows 98, Windows

2000, Windows ME, Windows NT, Windows CE, Linux, Unix, BSD, MacOS 9.x, MacOS X, Sun OS, PALM, or a Java-based operating system, among others.

The programmable circuitry 114 may take various forms. These forms may enable a user to program the client machine 110 using various interfaces such as a keyboard, mouse, network, drive, and handheld circuitry, among others.

The network interfaces may take various forms. These forms may include various circuitry for communicating through ethernet, wireless ethernet, Blue Tooth, phone lines, and modems, among others.

User interfaces may take various forms. These forms may include monitors, keyboards, wireless devices, handheld devices, and a mouse, among others.

The storage mediums 120 may take various forms. These forms may include hard drives, floppy drives, removable drives, cards, CD-ROM, CD-RW, CD-R, DVD, DVD-R, DVD-RW, RAM, and flash memory, among others.

The storage mediums 120 may store various applications 122, applets 126 and or data 128. The client 110 may function, for example, to access, display and manipulate data stored on a server and associated with network appliances. The client may use installed applications to access, display and manipulate the data. Alternately, the client may download applications, applets, and object classes, among others, to access, display, and/or manipulate the data. Furthermore, the client may use various combinations of installed and downloaded application, applets, object classes, among others.

The applications, applets, object classes may take various forms. These forms may include internet browsers, stand alone applications, interpreters, libraries, and instruction sets, among others.

In one exemplary embodiment, the client may connect to a server through a network interface 116. The client may have a JAVA enabled web browser. The web browser may function to acquire an applet from the server through the network interface 116. The applet may function to enable access to the data, display the data in various forms, and enable manipulation of the data. The client may manipulate data on the server to alter map configurations, network appliance associations, accessibility and permission information, annotate data associated with events, and network application configuration data, among others.

Further, the applet or applets may also function to permit changing and/or manipulation of configuration data associated with network appliances. For example, one or more parameters associated with one or more network appliances may be changed. A parameter associated with several network appliances may be changed to a same value for each network appliance. Alternately, a single value may be changed associated with a single parameter of a single network appliance. Furthermore, configuration settings may be uploaded to the server for future implementation on the network appliances.

The applet or applets may enable the client machine to display data. For example, the applet or applications may display a map. The map may have icons associated with the network appliances. Further, these icons may be used to display representations of the data. These icons may also be superimposed on a graphic, image, map or plot, among others. Further, the icons may be arranged according to type, location, alarm state, configuration, parameter value, or organization, among others. Alternately, the applications or applets may display the data as a table. For example, the table may display a current value of a parameter associated with a sensor on or connected to a network appliance. Alternately, the table may display alarm states associated with network appliances. Further, the table may display configuration parameters and data associated with network appliances. The table may further enable manipulation and

changing of the values within the table. Alternately, the data may be displayed in graphical forms. These graphs may additionally offer the ability to chart data associated with one or more sensors associated with one or more network appliances. However, various other display methods may be envisaged. The applications or applets may also function to dynamically download data objects, classes, program elements, useful for accessing, displaying and/or manipulating new data elements. For example, a network class loader may be implemented in an application or applet such that new data classes may be implemented. These may, for example, be written in JAVA.

The applications and/or applets may also function to display image data. The image data may, for example, be associated with events, network appliances, and sensor data, among others. The applet or applets may display the image data in association with the events, network appliances, and/or sensor data.

In one exemplary embodiment, the client machine 110 may be a personal computer running an operating system such as, for example, Windows 2000. The client machine 110 may have an browser such as Internet Explorer 6.0 and be Java enabled.

In another exemplary embodiment, the client machine may be a handheld device with an operating system such as PALM or WINDOWS CE and be Java enabled. However, various devices may be envisaged. In addition, various operating systems and computer languages may be used.

In this manner, a client machine 110 may have fully functional access to information stored on the server and associated with network appliances. Further, the client may function to view, create, and manipulate groupings of network appliances. The client machine 110 may function to establish permissions to groupings.

Figure 5 is a block diagram of an exemplary embodiment of a server as seen in Figure 1. A server 130 may have a processor 132, programmable circuitry 134, network interfaces 136, and storage mediums 138 and user interfaces 148. A storage medium 138 may hold databases 140, applications 142, instructions 144 and map configuration data 146. However, these element may or may not be included. Further, these elements may be separate, together, or in various combinations, among others.

A processor 132 may take various forms. These forms may include CPUs, embedded processors, JAVA enabled processors, and various computational circuitry, among others. Further the processor 132 may operate using an operating system such as Window 2000, Windows NT, Linux, BSD, UNIX, Mac OS X, Mac OS 9.x, or a Java-based operating system, among others.

A programmable circuitry 134 may take various forms. These forms may enable a user to program the server 130 using various interfaces such as a keyboard, mouse, network, drive, and handheld circuitry, among others.

A network interfaces 136 may take various forms. These forms may include various circuitry for communicating through ethernet, wireless ethernet, Blue Tooth, phone lines, and modems, among others.

Storage mediums 138 may take various forms. These forms may include hard drives, floppy drives, removable drives, cards, CD-ROM, CD-RW, CD-R, DVD, DVD-R, DVD-RW, RAM, and flash memory, among others.

The storage mediums 138 may hold databases 140, applications 142, instructions 144 and map configuration data 146. The databases 140 may take various forms. These forms may include Oracle databases, SQL compatible databases, Jet databases, generic databases, tables, and spreadsheets, among others. The map configuration data 146 may also be stored in a

database 140. The instructions 144 may take various forms. These forms may include compiled code, interpreted code, Java code, Visual Basic code, C++ code, HTML code, PHP code, and Perl, among others.

The user interfaces 148 may take various forms. These forms may include monitors, keyboards, wireless devices, handheld devices, and a mouse, among others.

The server may function to download data from network appliances through the network interfaces 136. The data may, for example, be stored in the databases 140. This data may be sensory data, configuration data, image data, among others. Further, the server may include applications and instructions for communicating with the network appliances.

A server 130 may also function to communicate with one or more client machines through the network interface or interfaces 136. The server 130 may transfer applications 142 to the client machine. These applications and instructions may enable the client machine 110 to retrieve, display, and/or manipulate data. These applications may also be delivered in parts, classes, or software objects on an as needed basis.

In one exemplary embodiment, a client machine may request an application from the server. The server may deliver at least part of the application to the client machine. For example, a browser on the client machine may request a Java applet. The Java applet may enable the client machine to access, display and manipulate data. For example, the applet may enable the client to organize and group network appliance data, develop user groups, change user access information, display maps, manipulate icons and map features, change network appliance configurations, display alarms, and annotate data, among others. Further, the client machine may store information on the server.

For example, the server may deliver an application enabling the client to access the database and display image data associated with a camera enabled network appliance.

Alternately, the server may deliver a part of an application enabling the client to display a table of network appliances and their associated parameters such as a value of a sensor or an alarm state, among others. Further, the server may deliver a part of an application which displays a tree of network appliances associated into groups.

The server may also deliver an application and associated map configuration data. The application may enable the client to access and display a map. The map may have icons superimposed on a background image. The icons may represent network appliances or groupings of network appliances. Further, the icons may link to present or historical values of the network appliances associated with the icons. In addition, an action such as clicking an icon may initiate another display such as another map, table, or graph. The icons may have an appearance indicative of type, capabilities, status, alarm state, present or historical value of a parameter or sensor output, or responsible party, among others. The icons may be arranged in a manner indicative of physical location, type, capabilities, status, alarm state, present or historical value of a parameter or sensor output, or responsible party, among others. Moreover, the background image may be a picture, video image, graph, contour plot, and vector plot, among others. The application may also enable the client machine to manipulate user access data stored on the server. The application may also enable the client machine to store map configuration data on the server 130.

Figure 6 is a block diagram of a network appliance, for use in the system as seen in Figure 1. The network appliance 150 may have a processor 152, a programmable circuitry 154, one or more network interfaces 156, one or more storage mediums 158, and one or more sensors 162, among others. The storage medium 158 may hold data 160, among others. However, these elements may or may not be included. Further, these elements may be separate, together, or in various configurations, among others.

The processor 152 may take various forms. These forms may include CPUs, embedded processors, JAVA enabled processors, and various computational circuitry, among others.

The programmable circuitry 154 may take various forms. These forms may enable a user to program the network appliance 150 using various interfaces such as a keyboard, mouse, network, drive, and handheld circuitry, among others.

The network interfaces may take various forms. These forms may include various circuitry for communicating through ethernet, wireless ethernet, Blue Tooth, phone lines, and modems, among others. Further, the network interface may enable the network appliance to connect to various networks including global networks, LANs, WANs, phone networks, page networks, satellite communication systems, and wireless networks, among others. The network interface may enable communication between the network appliance 150 and a server and/or other network appliances. Further, the network interface may enable the use of various methods, protocols, and standards, included HTTP, FTP, SNMP, TCP/IP, LDAP, and others.

The storage mediums 158 may take various forms. These forms may include hard drives, floppy drives, removable drives, cards, CD-ROM, CD-RW, CD-R, DVD, DVD-R, DVD-RW, RAM, and flash memory, among others. Further, the storage medium may store data associated with network appliance configuration, sensors, user access, other network appliances, and algorithms, among others.

The sensors 162 may take various forms. These forms may include temperature sensors, pressure sensors, airflow sensors, alarm sensors, dry contact sensors, humidity sensors, cameras, video cameras, infrared cameras, power quality sensors, data traffic sensors, acoustic sensors, and motion sensors, among others.

The network appliance 150 may function to communicate with the server. The communication may, for example, take the form of a ping, an HTTP GET, an HTTP POST, a

SNMP message, an email message, or an FTP command, among others. With the communication, the network appliance may upload data, download configuration and/or accessibility settings, download program information, and indicate status. The communication may also use various security protocols and methods. Alternatively, the network appliance 150 may communicate with another network appliance acting as an intermediary between the server and the network appliance 150. As such, the information above may be exchanged between the network appliance 150 and the other network appliance acting as the intermediary. In both cases, the network appliance may deliver data on a schedule, as it is available, in response to a request, in response to an alarm, or in other manners. Further, the data may be formatted in various protocols including HTTP or FTP, among others.

The network appliance 150 may also communicate with other network appliances in a cluster. The cluster of network appliances may use various means for communication including HTTP, SNMP, and FTP, among others. The cluster may also establish relationships, a directory, and share resources, among others.

In one exemplary embodiment, the network appliance may collect image data in response to an open door alarm or motion alarm. The network appliance 150 may then upload the data to a server. The server may then provide the image and the alarm data to a client machine.

In another exemplary embodiment, a client machine may request temperature data from the server, the server may collect the data from the network appliance 150. The server may then forward the data to the client machine.

In a further example, the client machine may alter configuration data. The data may be stored on the server. The network appliance 150 may retrieve the configuration data from the server and adapt.

Turning to methods of displaying and manipulating data, a map configuration may be established and stored on the server. The map configuration may be accessible by various user. In one exemplary embodiment, Figure 7 is a schematic block diagram of a user association for the map configuration. A first user 172 may create a mapping of icons. The icons may be associated with network appliances. These network appliances may be active or passive devices. Further, the icons may be arranged and/or superimposed on a background image. The first user may establish a permission data. The permission data may for example give a second user 176 access to the map data 174. The second user may be given permission to view or edit the map configuration data, or both. Alternately, the first user may give viewing permission or exclude another user 178.

Additionally, the map view may be "locked" or "unlocked". When "locked", the icons and objects on the view are not movable, preventing accidental or intentional manipulation of the layout. The privilege of "unlocking" of the map view can be restricted, allowing a map to be created and maintained by one user account, and safely shared with other, less privileged, users.

The icons may take various forms. These visual forms may be indicative of type, alarm status, parameter value, capabilities, and version, among others. For example, an icon may have a shape representative of it capabilities, a color representative of a sensor value, a right hand flag with a label, a top flag with a numerical value. In addition, the flags may change color in response to alarm conditions. However, various changes and uses of visual characteristics can be envisaged to represent various data associated with network appliances. Each icon may have some, all, or none of these features.

The icons may also link to other images, displays, and data. For example, the user, through an action such as, for example, clicking on the icon may display another mapping, a

data table, and an icon configuration, among others. Furthermore , the user may manipulate the icon configuration and store the configuration on the server.

Further, the icons may be arranged in a display in accordance with some characteristic. For example they may be arrange according to a sensor value, an alarm state, a physical location, or randomly, among others. Figure 8A is schematic block diagram of an exemplary embodiment of a map. The icons may be arranged in a display area. For example, icons associated with a user may be viewed. Figure 8B is a schematic block diagram of an exemplary embodiment of a map. As shown, the icons may be arranged according to an alarm state as indicated by a shaded flag. Alternately, the icons may be arranged according to physical location as shown in Figure 8C. For example, the location may be a location within a room, geography, or server rack. Further, the icons may be superimposed on a map or image indicative of the location. The map or image may change in response to events associated with the network appliances. For example an image representing a room may be replaced with a similar image indicating an open door. However, the image may be a picture, video image, plot, graph, blueprint, or map, among others. In another example, the icons may be arranged according to network appliance type, as depicted in Figure 8D. The shape of the icon may for example represent the type or version. However, various pairings between visual characteristics and data may be envisaged. These map configurations and associated accessibility information may be stored on the server and accessed by the client.

The icons and object displayed on the map view may include both active network devices and passive devices. The ability to add and manipulate the passive devices along with the active network devices may allow the user to accurately represent the physical environment of his equipment rooms, for example. Other exemplary implementations may allow the end-user

to import graphical images in a variety of formats (GIF, BMP, JPG, etc) to use as icons customized for their specific equipment (both active and passive).

In one embodiment, a mapping may be associated with a grouping of network appliances. This grouping may, for example, be related to physical location or topology. In one example, environmental sensor readings may be displayed on the map views as part of the icon. The map view may display a single sensor attribute at a time on each of the active devices supporting the given sensor. For example, when temperature is selected, each device that supports a temperature sensor has the most current reading of that sensor presented. In conjunction with the physical representation afforded by the map view, this may enable a presentation of the two-dimensional "field" associated with the given sensor. The map view may also allow very rapid selection of different sensors readings via a context menu, allowing a user to quickly cycle between the values of different sensors without needing to open additional windows. For sensor types that have potentially different units of measurement (degrees C versus degrees F, ft/min versus meters/min), the view appropriately converts all sensor values to the unit of measurement most appropriate to the locale and preferences of the user, even when the data actually supplied by the different devices is natively in different units (degrees C from one device, degrees F from another).

The map may also use map colorization. Map colorization refers to the ability to use color to represent sensor readings for an environment. This can be as simple as putting the sensor reading of the device on the icon or changing the color of the icon to represent a sensor threshold range. Also, the background of the map surrounding the icons may look like a contour plot to display sensor readings from around the room.

Another implementation of present invention may include support for a variety of enclosures, such as equipment racks and cabinets, that will allow presentation of multiple

devices stacked vertically at the same location. Map Colorization of these enclosures will allow sensor reading to be presented with respect to vertical positioning, as well as horizontal. In addition, the vertical positions will enable the presentation on the standard Map View of sensors values for a given "slice" of the room (i.e. all temperature sensors at the top of the racks, the middle of the racks, or under the raised floor).

Additional use of the feature could allow the presentation of various attributes generated from multiple related sensors in the same enclosure. For example, each rack could be displayed with the temperature delta between the temperature reading of the cool air flowing into the rack versus the exhaust temperature.

The map view may also auto-sort by alarm severity. For example, environmental sensor alarms may be sorted to be displayed at the top of the map, followed by network connectivity alarms, and lastly by devices that are not in alarm state.

The display string for each icon may be user configurable to vertically display a customizable user-friendly "name" for each device. The devices that are red may have environmental sensor alarms, the devices that are yellow may have network connectivity alarms, and the gray devices may be in a normal state. The colors may be user customizable. In the colorized mode, the display string may show the alarm status.

This ordering and representation allows the user to quickly determine which devices need attention, even in a group containing hundreds or thousands of devices, since the user can quickly look at the first devices listed and know which devices need attention. Also, the user can quickly conclude by the fact that the first device listed has no errors that none of the other devices currently do.

Network appliance configuration data may also be stored on the server. This configuration data may include parameters, contact data, alarm settings, email lists, alert lists,

algorithms, password and access data, and threshold data, among others. The client may retrieve and manipulate the data. For example, the client machine may display configuration data for multiple network appliances. This configuration data may, for example, be displayed as a table. The client machine may change some, all, or none of the data. Further, the client machine may change the value of a similar parameter associated with several network appliances to the same value.

Figure 9A and 9B are block diagrams depicting exemplary embodiments of tables for use in manipulating configuration data. Configuration data may take various forms. These forms may include parameters, settings, notification lists, address, responsibility lists, algorithms, software, and communications protocols, among others. The configurations may be changed by selecting a single cell and making a change as seen in the data column for network appliance #5. Alternately, all cells may be selected at once and changed to the same value as seen for parameter #1. However, fewer cells may be selected and changed as seen for parameter #2.

The system may provide a mass configuration mechanism for managing the settings for our HTTP configurable appliances. The server may be modified via its software plug-in architecture to handle mass configuration of any HTTP configurable appliance.

The use of HTTP Posts for configuration management of the appliances may allow configuration to be done more quickly, efficiently, and with better transactional integrity than SNMP-based configuration. Each HTTP Post may be used to configure multiple parameters in parallel, preventing the possibility of the appliance's configuration being partially updated (resulting in an unusable configuration) as well as reducing the number of network transactions required to complete a configuration update. As with all HTTP communications, the use of TCP/IP (as opposed to the UPD/IP used by SNMP) tends to minimize problems with WANs and firewalls.

The system may provide a mechanism for viewing the current software and hardware levels of our appliances and for updating them. A status column may display any errors that occur during the version query process and displays textual progress messages during the upgrade process. At any time, a user may click on a cell in this column and view a popup that displays the entire status message since they are frequently longer than can be conveniently displayed in a single column of a spreadsheet.

Another exemplary implementation of the system may include support for managing the configuration of an appliance that is only capable of communicating with the Server, but which cannot be communicated with by the Server. The system may store and maintain a copy of the settings desired by the user for a given appliance. These settings may be determined in the same fashion as they are currently set with the Mass Configuration interfaces. Since the Server cannot initiate communications with the firewalled appliances, it will simply record the desired settings to be communicated later.

On the appliances, support may be added for configuring the appliance to issue periodic HTTP Posts to the present invention, querying for configuration updates. Whenever a configuration update request Post is issues to the server, the server may have the option of including (as the content of the reply to the Post) a single command block structured the same as the input for an HTTP GET or POST would be if issued directly to the Appliance. When the reply to the configuration update request includes this content, the Appliance may process it as if the given HTTP GET or POST had been issued to the Appliance as normal. When completed, the output of this request may be delivered to the server as the input to a second HTTP Post to the present invention. The server may then process the input of the Post (which is the reply to the request he issued with the previous Post), and may either reply with the input for the next

HTTP GET or POST (repeating the process), or with no input (if no further requests are pending).

Figure 10 is a block flow diagram of an exemplary method for manipulating configuration data, among other data. In the method 190, the client machine may retrieve the data from the server, as seen in a block 192. A user may manipulate the data as seen in a block 194. Next, the server may store the data. Then, the data may be transferred to the network appliance or an intermediary network appliance as seen in a block 198.

For example, a client machine may request using an HTTP command data and/or applications associated with manipulating configuration data. A user may manipulate the data in a Java enabled browser. The client machine may then, using an HTTP command send the manipulated data to the server for storage. A network appliance may ping the server. The server may respond to the ping with the manipulated data. However, various means and protocols may be envisaged for performing the method 190.

Figure 11 is a block flow diagram of an exemplary method for use by the system. As above, the server may store configuration data as seen in a block 212. The network appliance or an intermediary may ping the server. This ping may take various forms and use various protocols. For example, the ping may take the form of an HTTP POST, HTTP GET, or FTP command, among others. The sever may respond to the ping as seen in a block 216. The server may transfer data using a security protocol such as SSL. If configuration data or an upgrade is available, the server may include the data with the response. The network appliance may then respond to indicate the transfer was successful. Subsequently, the server may respond with more data or an indication that no more data is available.

An example of this interchange might go as follows:

An appliance issues its periodic HTTP Post for requesting configuration updates:

POST /centra/configquery HTTP/1.1

Host: 192.168.1.218:81

User-Agent: NetBotz/1.1.3

Accept: */*

Accept-Encoding: gzip

Accept-Language: en

The server determines that there is a pending configuration update, and issues a POST to set the new setting (in this case, enabling the temperature threshold for a low of 60 degrees and a high of 80 degrees):

HTTP/1.1 200 OK

Date: Mon, 06 Aug 2001 17:15:22 GMT

Server: Apache/1.3.17 (Unix) PHP/4.0.4pl1

Last-Modified: Thu, 02 Aug 2001 15:27:24 GMT

Content-Length: 260

Content-Type: binary/x-user-request

POST /setTemp HTTP/1.1

Authorization: Basic bmV0Ym90ejpwYXNzd29yZA==

Host: bc10

User-Agent: USER AGENT/1.2

Accept: *.*

Accept-Encoding: gzip

Accept-Language: en

VARIABLE=VALUE

The Appliance may process the output of its Post as if the command had been sent to it through its web server, modifying the settings and generating a reply. The Appliance then issues another configuration request to the present invention, delivering the output:

POST /centra/configquery HTTP/1.1

Host: 192.168.1.218:81

User-Agent: AGENT/1.1.3

Accept: */*

Accept-Encoding: gzip

Accept-Language: en

HTTP/1.1 200 OK

Date: Mon, 06 Aug 2001 17:15:25 GMT

Server: SERVER/1.1.3

Last-Modified: Mon, 06 Aug 2001 17:15:25 GMT

Content-Length: 200

Content-Type: text/plain

VARIABLE 1: VALUE 1

VARIABLE 2: VALUE 2

The server may process the output, and determines if another request needs to be issued. If so, it replies to the Post with the next request (repeating steps 2-4 until all requests are done). If not, it simply replies with no output:

HTTP/1.1 200 OK

Date: Mon, 06 Aug 2001 17:15:27 GMT

Server: Apache/1.3.17 (Unix) PHP/4.0.4pl1

Last-Modified: Thu, 02 Aug 2001 15:27:26 GMT

The Appliance sees that there are no further requests, and waiting until the next configured configuration update polling time before issuing another configuration request HTTP Post.

This mechanism may allow the full span of features accessible through the Appliance's web server (including those provided via add-ons) to be accessed without requiring custom coding or modification, since any HTTP GET or POST request can be wrappered as shown above.

In addition, this approach may allow appliances that have lost their configuration to be configured simply by pointing them at the system. This Feature may use a set of HTTP GETs through this mechanism in order to validate an Appliance's configuration before applying any needed changes.

Network appliance data may be organized and associated by various means. The data may be organized by location, responsible party, organization, network appliance type, version, alarm state, and status among others. For example, the network appliances may be organized

into groups. Groups may be dynamic or static lists of appliances that represent a set of appliances. Each logical group may be implemented through SQL query (used to produce a list of appliances) or a specific list of appliances, and a list of users that have access to the group for security, among others. When an SQL query is defined for a group, appliances are automatically assigned to the group (as well as removed from the group) based on their attributes matching the conditions dictated by the SQL query. For example, a group may be defined by an SQL query which logically selected "all appliances where the application version = 1.2". Appliances upgraded from application version 1.1 to 1.2 would automatically be added to the group, while appliances upgraded from version 1.2 to 1.3 would automatically be removed.

Groups may be used to display the hierarchy of a business organization, the responsible IT person for said group, or to represent a physical location in a building. Other exemplary implementations may allow matching on a wide variety of attributes, including current sensor readings, alert states, and custom, user-defined appliance attributes.

Further the associations may be manipulated and changed. For example, the client machine may retrieve data associate with network appliance and display them in a tree. Figure 12 is a schematic of an exemplary tree. The tree may associate appliances in various groups and give access to individual appliances on various levels of the tree structure. Moreover, a user may manipulate groups, appliances associated with the groups, user access and permissions associated with the network appliance and groups, and the tree visual characteristics, among others.

The groups may also be represented in a table. The table may display various data associated with the network appliances. Further the table may be updated as data changes on the server. This update may for example, occur following a ping and/or query to the server. Alternately, the update may be provided by the server.

Figures 13A, 13B and 13C are block diagrams depicting an exemplary table associated with an exemplary group. In this exemplary embodiment and appliance may have an on/off status. This status may change as seen with network appliance #1. Further the other visual indications may be used to indicate status, such as, for example, shading. In addition, visual and/or acoustic indicators may be used to indicate alarms or valuations relative to thresholds. For example network appliance 2 may have an alarm associated with value #1 and shade that cell. The cells may be colorized to indicate alarm state. A user may also manipulate the values, parameters, and other characteristics displayed in a table. For example, a user may display the model or version. Further, the order of the network appliances may be varied in association with an alarm, a data value, or grouping.

A table may also be used to display historical data of events, alerts and alarms, among others. The data may include a data, time, available images, and other data.

Data associated with network appliances may also be displayed as a graph as seen in Figure 14. The graph may display the same type of data for several network appliances, various data from various sensors for the same appliance, or various combinations, among others. The graph may be composed of historical data or may be updated as new data is available. Further, the graph may replay data, changing the graph to represent a next value in a series of values according to an accelerated schedule.

To compact the amount of data the server stores overall, a schema may be implemented to only store the changes in the environment. For example, if the system collected data from an Appliance every 10 minutes, and the temperature of the room was constant for over an hour creating a data point for each collection interval may increase the size of the stored data. Instead, only the changes may be recorded so the environment can be played-back to the user in as efficient a manner as possible. Since most environmental sensors tend to change value slowly

and infrequently, this enables a significant reduction in the amount of data stored in the database of the present invention without any loss of resolution and accuracy: storing 100 rows, 1 per minute, indicating the same temperature reported by the same sensor is no more accurate or detailed (but consumes significantly more data) than one row reporting that the sensor was a given temperature for the 100 minutes between two points in time. This compaction of the recorded sensor data enables significantly more data to be recorded for more appliances for a longer time (estimates are 20-100 times as much as a conventional 1 sample per row schema). Each row may include both a starting timestamp and an ending timestamp, allowing easy creation of SQL queries requesting sensor readings at any given time (i.e. SELECT * WHERE ((START_TIME <= T) AND (END_TIME >= T));).

The graphs may be depicted based on a time range and a set of particular sensor readings. Allowing more than one appliance to be graphed at a time allows users to physically view the patters of environmental changes as well as compare one area of a location against another. The graphs themselves may be organized by day, week, month, or for the entire time range provided. These graphs may then be saved as in a graphic format, such as, a JPEG, GIF, or BMP file, among others, for email and/or reports, or can be exported as comma-delimited text to another utility of the users choosing.

The graphs may also include markers indicating any alerts associated with the displayed sensor on the selected appliances. These markers may appear on the line graph at the point in time where the alarm was reported or on an axis, among others. Different markers may be used for alarms reporting errors versus alarms reporting the return-to-normal of a previously out-of-bounds sensor reading For example, a solid bullet may be used for errors, and an open bullet for return-to-normal alarms. This feature allows a concise and comprehensive view of the history of

a given sensor on a set of appliances, both including the recorded data and highlighting the important events associated with that history.

Since some environment changes can be radically different than others, the graph view may implement zooming in on a particular set of data points. This provides the user with a more detailed graph of a smaller time range. Just like the other graphs, a zoomed-in graph can then be saved to a graphic format for email or exported as a comma delimited file for use in another application.

When the graph zoom is activated, the time and sensor units scales may be appropriately recomputed based on the selected range. In addition, the legend associated with the graph may be reduced to just include those appliances that have sensor data contained within the zoom window, allowing the zoom view to be effectively used to pull detailed information out of a graph containing more lines of data than could typically viewed effectively.

In addition, data may be displayed and/or manipulated in other formats. For example, Figure 15 shows a display for image data. The display area 230 may show an image. The image may be associated with an event such as, for example, a door sensor, an alarm, or a specified time, among others. The image 136 may be displayed with event data 232 and/or appliance data 234. However, more than one image may be displayed. A series of images may be displayed from a single appliance. Alternately an array of images from several network appliances.

The event may be a recent event or a stored historical event. Further, the images and data may be stored in a manner which associates the image with the event and/or the data.

As such, a remote monitoring system is described. In view of the above detailed description of the present invention and associated drawings, other modifications and variations will now become apparent to those skilled in the art. It should also be apparent that such other

modifications and variations may be effected without departing from the spirit and scope of the present invention as set forth in the claims that follow.

WHAT IS CLAIMED IS:

1. A system, the system comprising:
at least one network appliance;
a server, the server communicatively coupled to the at least one network application, the server comprising:
means for storing sensor data gathered from the at least one network appliance;
means for storing instructions operable to enable the client machine to submit requests for the sensor data, to display the sensor data, and to manipulate the sensor data; and
means for transferring the instructions to the client machine; and
a client machine, the client machine communicatively coupled to the server, the client machine comprising:
means for acquiring instructions from the server, the instructions operable for requesting from the server data associated with one or more network appliances, the instructions operable for displaying the data associated with one or more network appliances, and the instructions operable for manipulating the server data associated with one or more network appliances; and
means for operating with said instructions.

2. The system of claim 1, the system further comprising:
the server further comprising:
means for storing configuration data associated with the at least one network appliance; and
means for transferring the configuration data to the at least one network appliance.

3. The system of Claim 2 wherein the means for transferring the configuration data is associated with a hypertext transfer protocol.

4. The system of Claim 2, the system further comprising:
the client machine further comprising
means for retrieving the configuration data from the server; and
means for manipulating the configuration data.

5. The system of Claim 1 wherein the means for acquiring instructions is associated with an internet browser.

6. The system of Claim 1 wherein the means for operating is associated with an internet browser.

7. The system of Claim 1 wherein the instructions as associated with Java.

8. A server, the server communicatively coupled to a client machine, the server communicatively coupled to at least one network appliance, the server comprising:
means for storing sensor data gathered from the at least one network appliance;
means for storing instructions operable to enable the client machine to submit requests for the sensor data, to display the sensor data, and to manipulate the sensor data; and
means for transferring the instructions to the client machine.

9. The server of claim 8 wherein the instructions comprise Java code.

10. The server of claim 8, the server further comprising:
means for storing configuration data associated with the at least one network appliance.

11. The server of claim 10, the server further comprising:
means for receiving a communication from the at least one network appliance; and
means for selectively responding to the communication with at least part of the configuration data.

12. The server of claim 11 wherein the communication is associated with a hypertext transfer protocol.

13. The server of claim 8, the server further comprising:
means for storing data associated with at least one map configuration, the at least one map configuration associated with one or more network appliances; and
means for delivering the data associated with the at least one map configuration to the client machine, the client machine interpreting the data associated with the at least one map

configuration to display one or more visual icons associated with said one or more network appliances.

14. The server of claim 8, the server further comprising:
means for storing at least one object class; and
means for transferring said at least one object class to the client machine, the client machine operable to interpret said object class, the object class operable to add functionality to the instructions.

15. The server of claim 14 wherein said at least one object class is associated with a network class loader.

16. The server of claim 8, the server further comprising:
means for dynamically grouping one or more network appliances and the data associated therewith.

17. The server of claim 16 wherein said instructions are operable to manipulate a grouping of one or more network appliances and the data associated therewith.

18. The server of Claim 16 wherein the grouping is associated with a responsible part.

19. The server of Claim 16 wherein the grouping is associated with a physical location.

20. The server of Claim 16 wherein the grouping is associated with an organization.

21. A client machine, the client machine communicatively coupled to a server, the client machine comprising:
means for acquiring instructions from the server, the instructions operable for requesting from the server data associated with one or more network appliances, the instructions operable for displaying the data associated with one or more network appliances, and the instructions operable for manipulating the data associated with one or more network appliances;

means for operating with said instructions.

22. The client machine of claim 21 wherein said means for acquiring and said means for operating comprise a Java-enabled browser.

23. The client machine of Claim 21 wherein the means for acquiring instruction is associated with a hypertext transfer protocol.

24. The client machine of claim 21 wherein said data is associated with configurations of one or more network enabled appliances.

25. The client machine of claim 24 wherein said manipulating said data comprises changing a parameter associated with several network appliances to a same value for each of the several network appliances.

26. The client machine of claim 21, the client machine further comprising:
means for acquiring data associated with at least one map configuration, the at least one map configuration being associated with one or more network appliances; and
means for interpreting the data associated with the at least one map configuration to display one or more visual icons associated with the one or more network appliances.

27. The client machine of Claim 26 wherein the one or more visual icons are displayed in ordered locations in a display area according to a state of the one or more network appliances.

28. The client machine of Claim 26 wherein the one or more visual icons are displayed in ordered locations representative of physical locations of the one or more network appliances.

29. The client machine of Claim 28 wherein the one or more visual icons are superimposed on a background indicative of the physical location.

30. A method for configuring more than one network appliances, the method comprises:

storing configuration data associated with the more than one network appliances on a device accessible by at least one of the more than one network appliances;

pinging the device accessible by the at least one of the more than one network appliances, the at least one of the more than one network appliances pinging the device; and

transferring at least part of the configuration data to the at least one of the more than one network appliances.

31. The method of claim 30 wherein the device and the at least one of the more than one network appliances are communicatively coupled through an interconnected network.

32. The method of claim 30 wherein the step of pinging is associated with an HTTP post method.

33. The method of claim 30 wherein the step of pinging is associated with an HTTP get method.

34. The method of claim 30 wherein the step of transferring is associated with an HTTP post method.

35. The method of claim 30 wherein the step of transferring is associated with an HTTP get method.

36. The method of claim 30, the method further comprising:

manipulating the configuration data with a client machine communicatively coupled to the device.

37. The method of 36 wherein the step of changing comprises:

manipulating a parameter associated with several network appliances to a same value for each of the several network appliances.

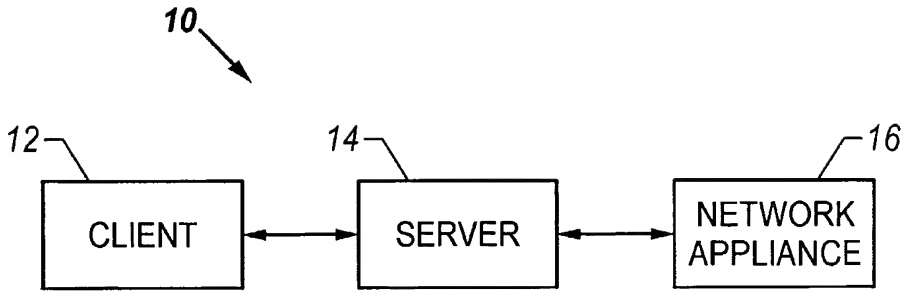

10
12
CLIENT
14
SERVER
16
NETWORK APPLIANCE

FIG. 1

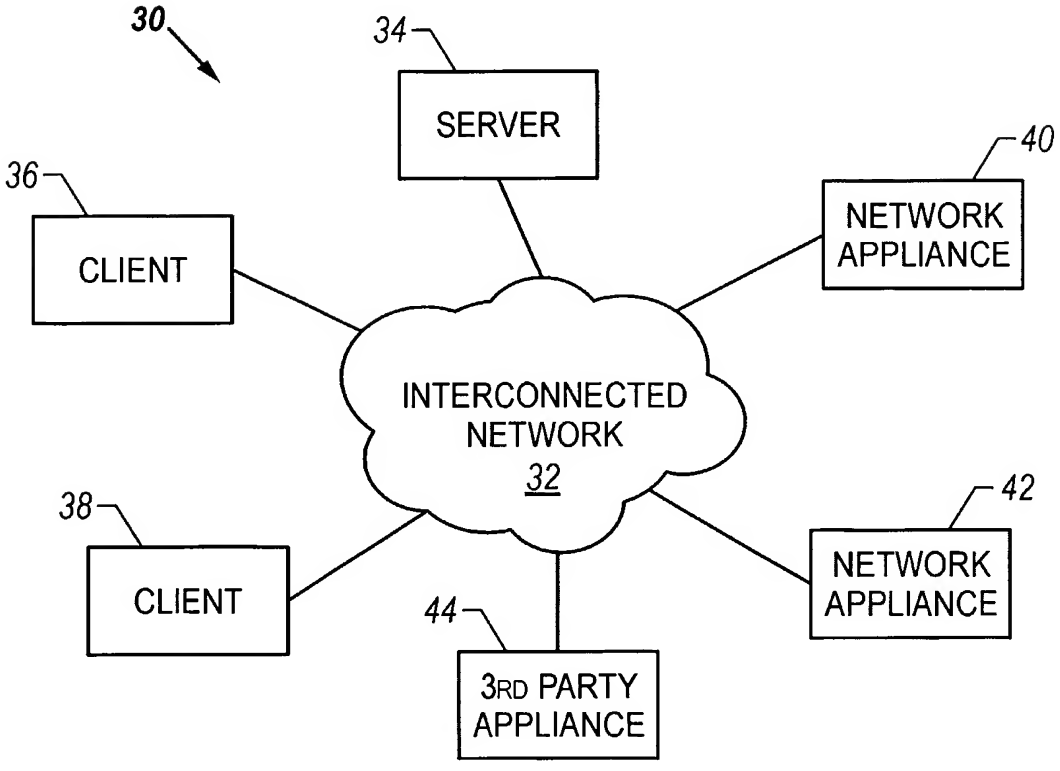

30
34
SERVER
36
CLIENT
40
NETWORK APPLIANCE
INTERCONNECTED NETWORK
32
38
CLIENT
42
NETWORK APPLIANCE
44
3RD PARTY APPLIANCE

FIG. 2A

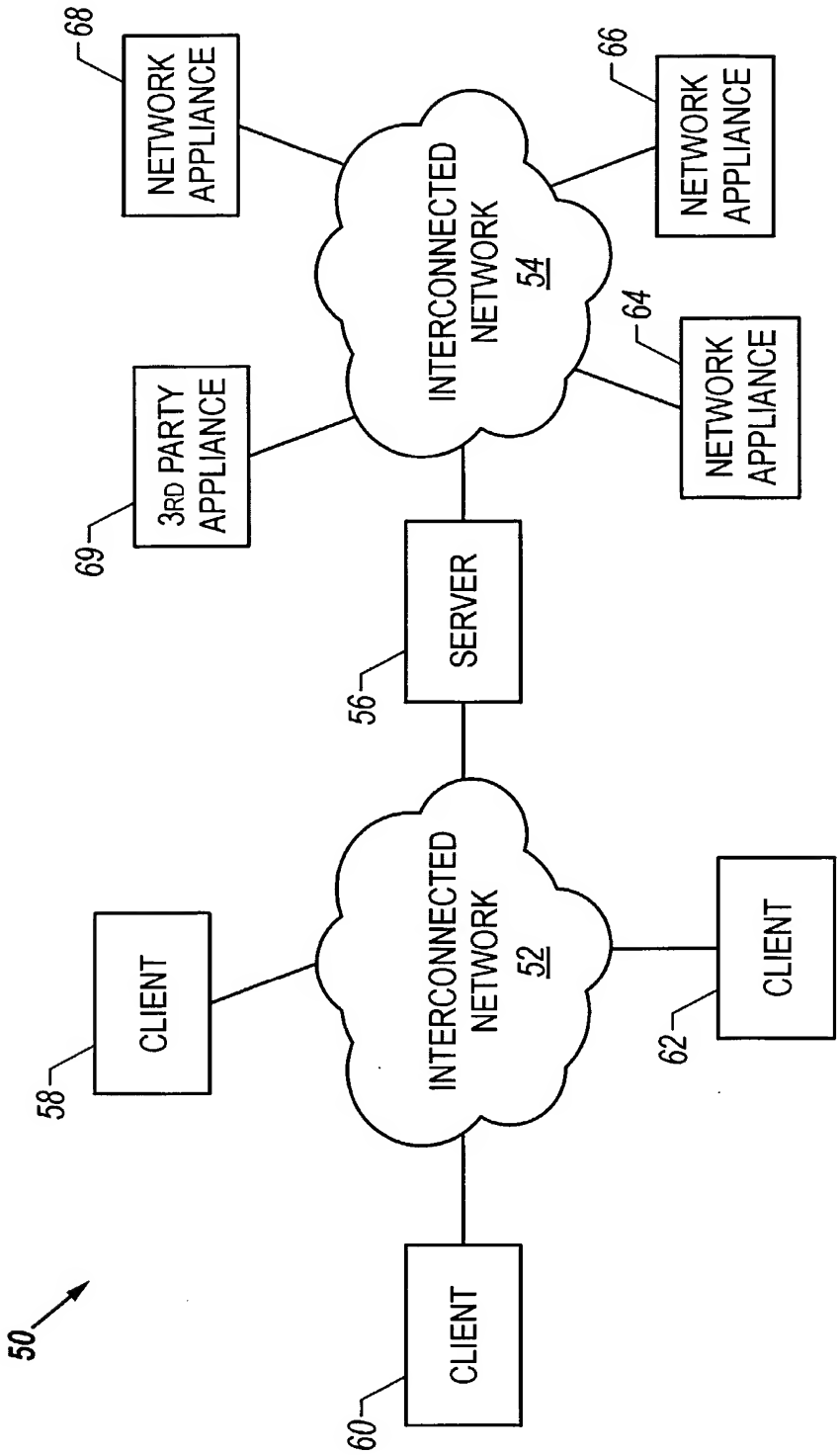

50
INTERCONNECTED NETWORK 52
INTERCONNECTED NETWORK 54
SERVER
56
CLIENT
58
CLIENT
60
CLIENT
62
NETWORK APPLIANCE
64
NETWORK APPLIANCE
66
NETWORK APPLIANCE
68
3RD PARTY APPLIANCE
69

FIG. 2B

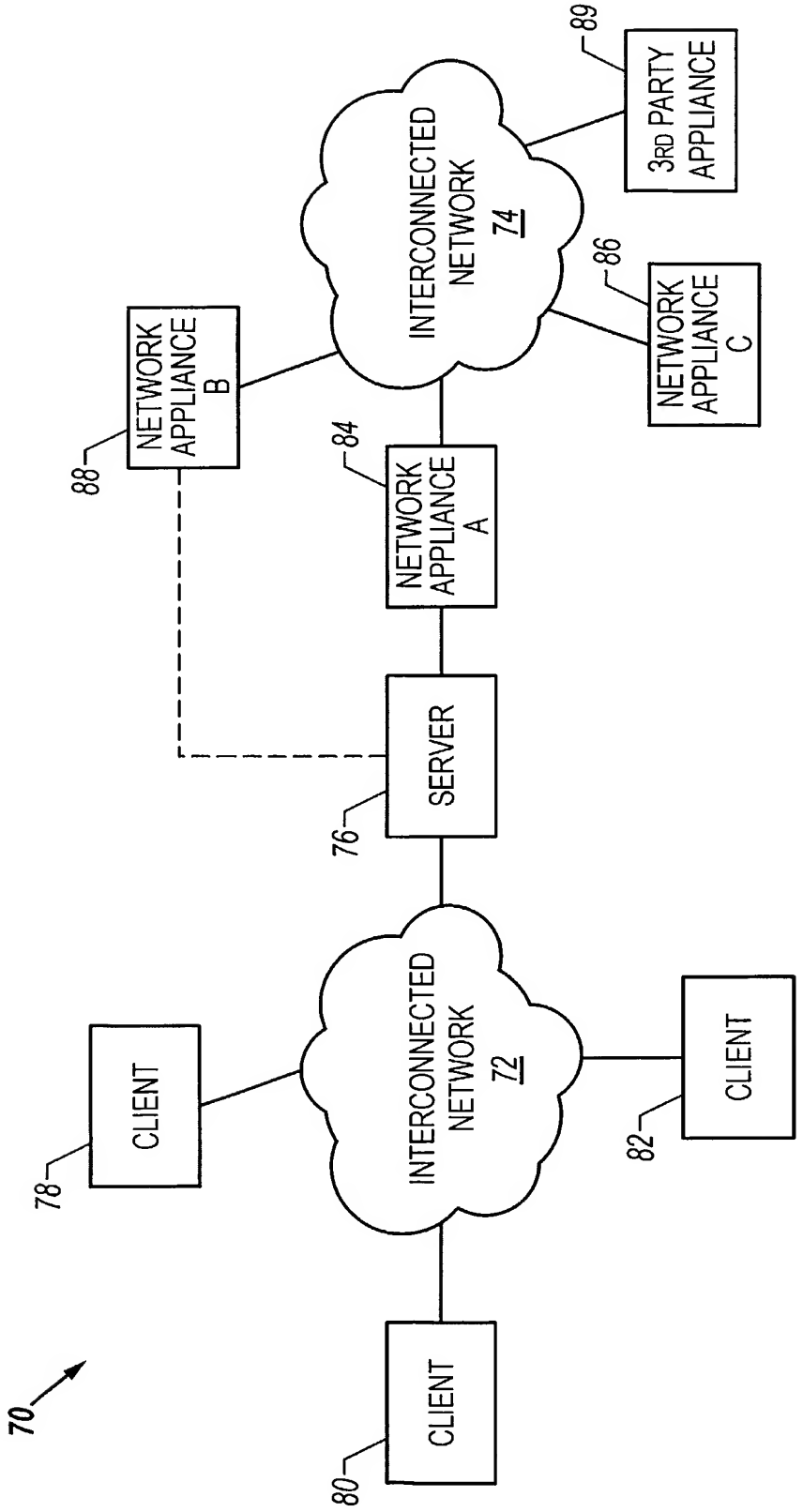

70
INTERCONNECTED NETWORK 72
INTERCONNECTED NETWORK 74
SERVER
76
CLIENT
78
CLIENT
80
CLIENT
82
NETWORK APPLIANCE A
84
NETWORK APPLIANCE B
88
NETWORK APPLIANCE C
86
3RD PARTY APPLIANCE
89

FIG. 2C

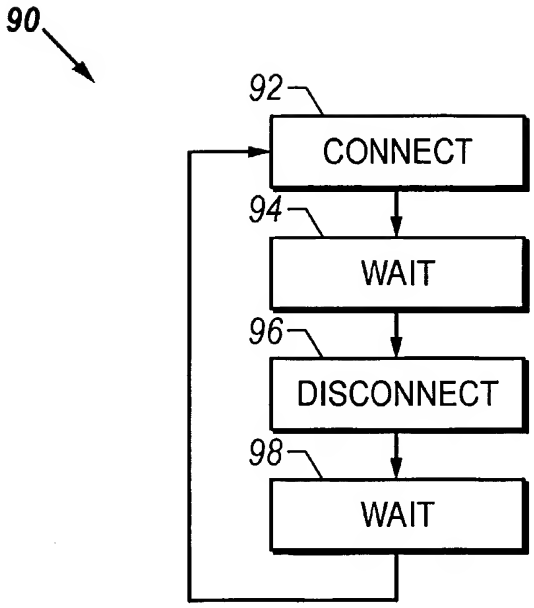

90
92
CONNECT
94
WAIT
96
DISCONNECT
98
WAIT

FIG. 3A

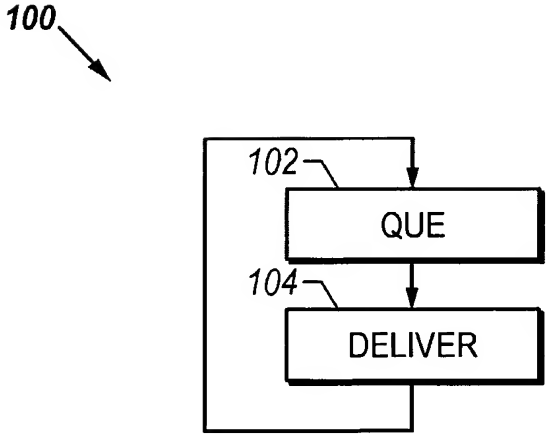

100
102
QUE
104
DELIVER

FIG. 3B

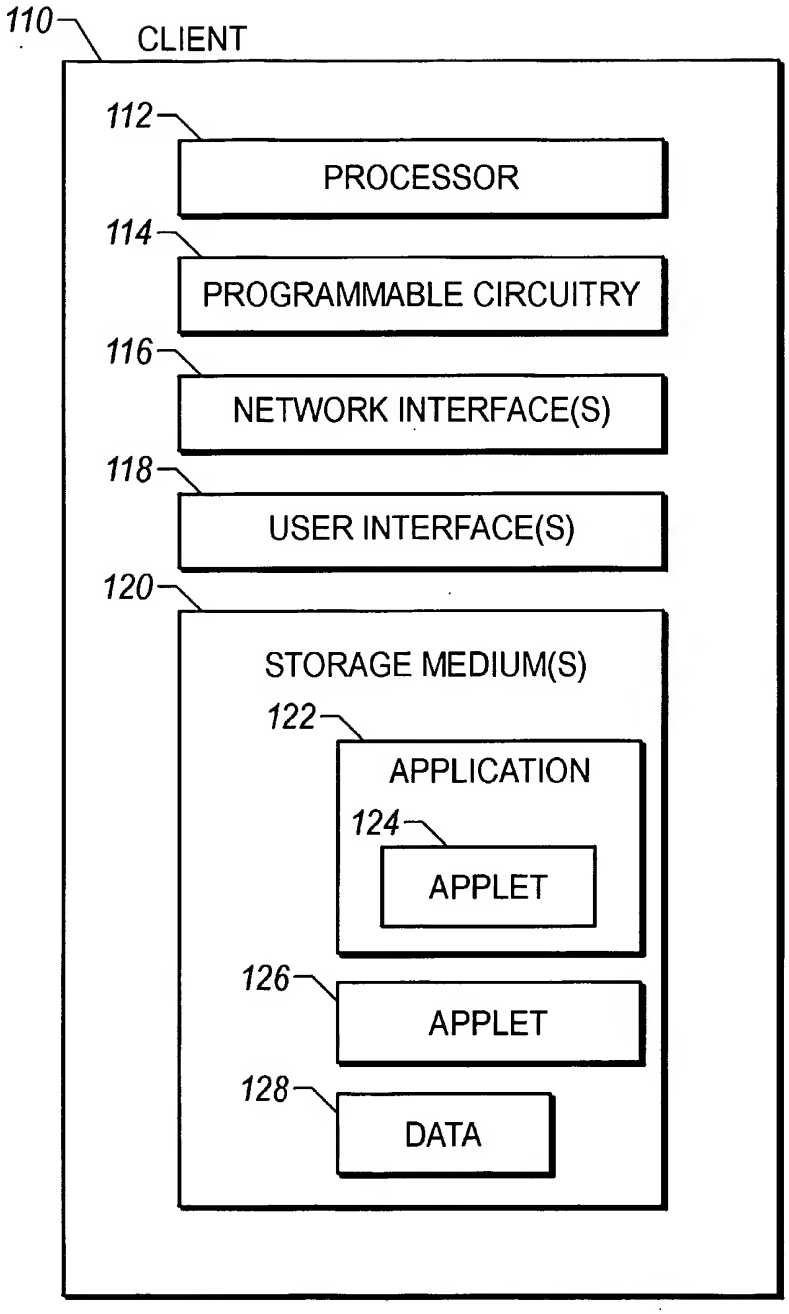

110
CLIENT
112
PROCESSOR
114
PROGRAMMABLE CIRCUITRY
116
NETWORK INTERFACE(S)
118
USER INTERFACE(S)
120
STORAGE MEDIUM(S)
122
APPLICATION
124
APPLET
126
APPLET
128
DATA

FIG. 4

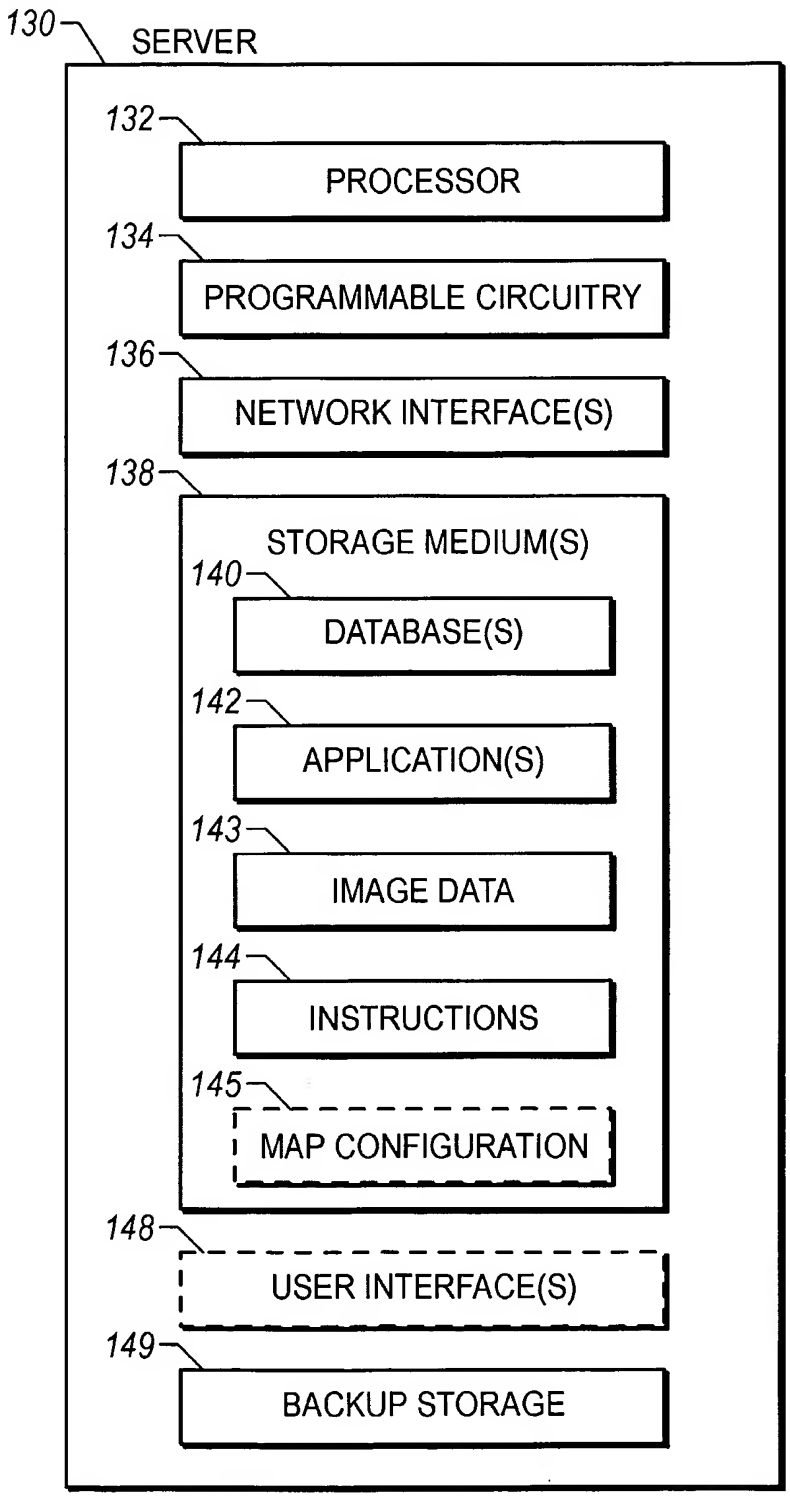

130
SERVER
132
PROCESSOR
134
PROGRAMMABLE CIRCUITRY
136
NETWORK INTERFACE(S)
138
STORAGE MEDIUM(S)
140
DATABASE(S)
142
APPLICATION(S)
143
IMAGE DATA
144
INSTRUCTIONS
145
MAP CONFIGURATION
148
USER INTERFACE(S)
149
BACKUP STORAGE

FIG. 5

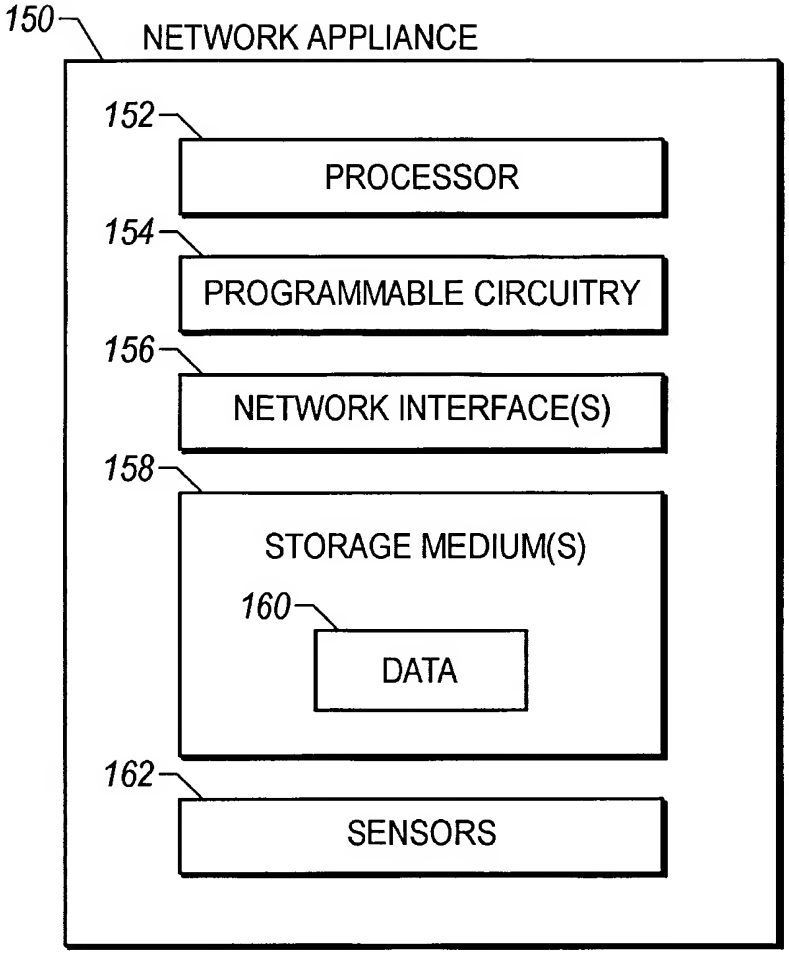

150
NETWORK APPLIANCE
152
PROCESSOR
154
PROGRAMMABLE CIRCUITRY
156
NETWORK INTERFACE(S)
158
STORAGE MEDIUM(S)
160
DATA
162
SENSORS

FIG. 6

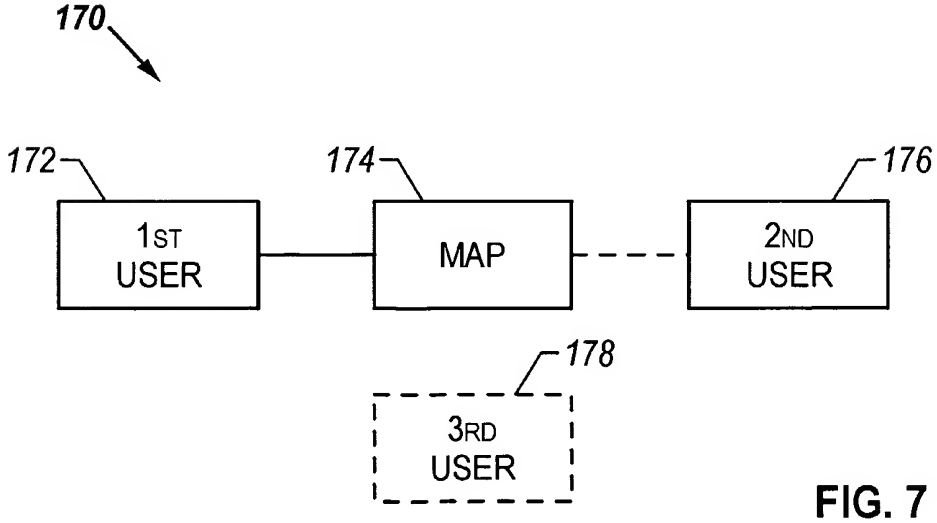

170
172
1ST USER
174
MAP
176
2ND USER
178
3RD USER

FIG. 7

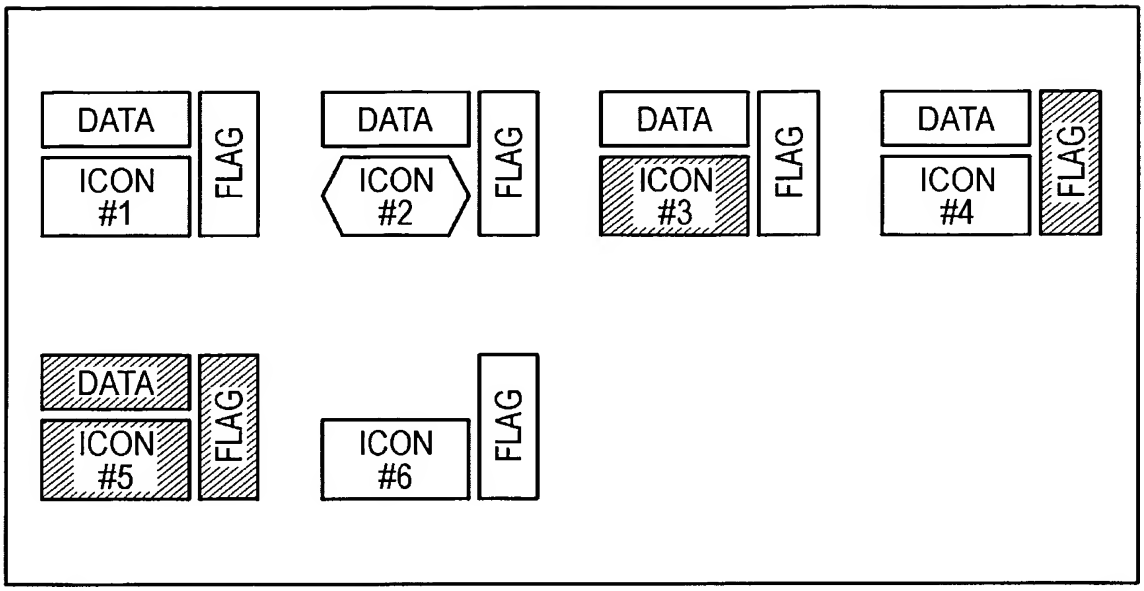

DATA
ICON #1
FLAG
DATA
ICON #2
FLAG
DATA
ICON #3
FLAG
DATA
ICON #4
FLAG
DATA
ICON #5
FLAG
ICON #6
FLAG

FIG. 8A

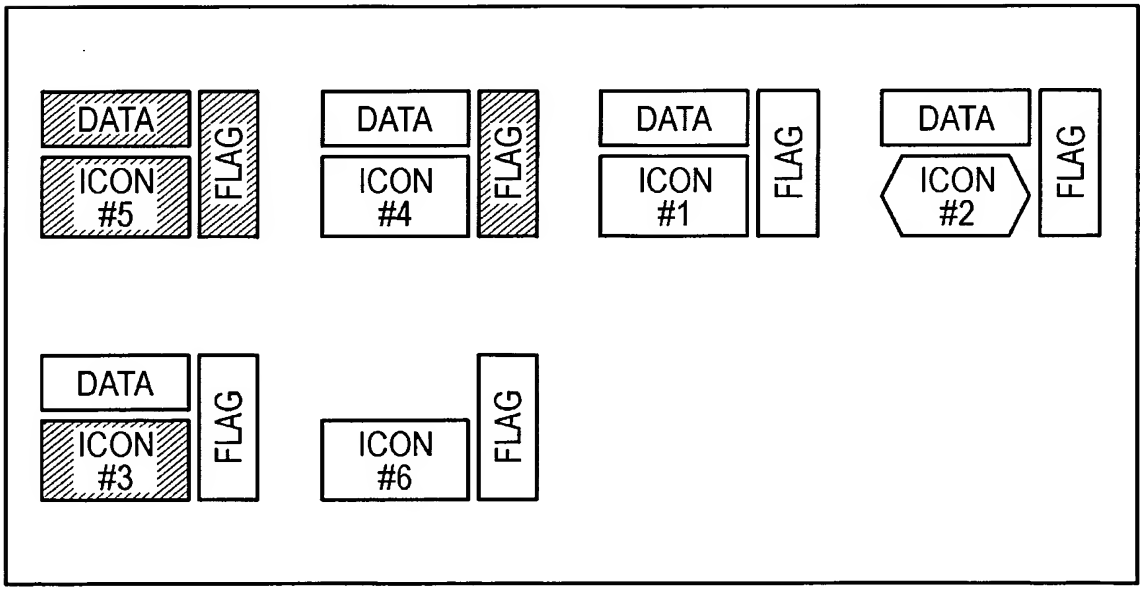

DATA
ICON #5
FLAG
DATA
ICON #4
FLAG
DATA
ICON #1
FLAG
DATA
ICON #2
FLAG
DATA
ICON #3
FLAG
ICON #6
FLAG

FIG. 8B

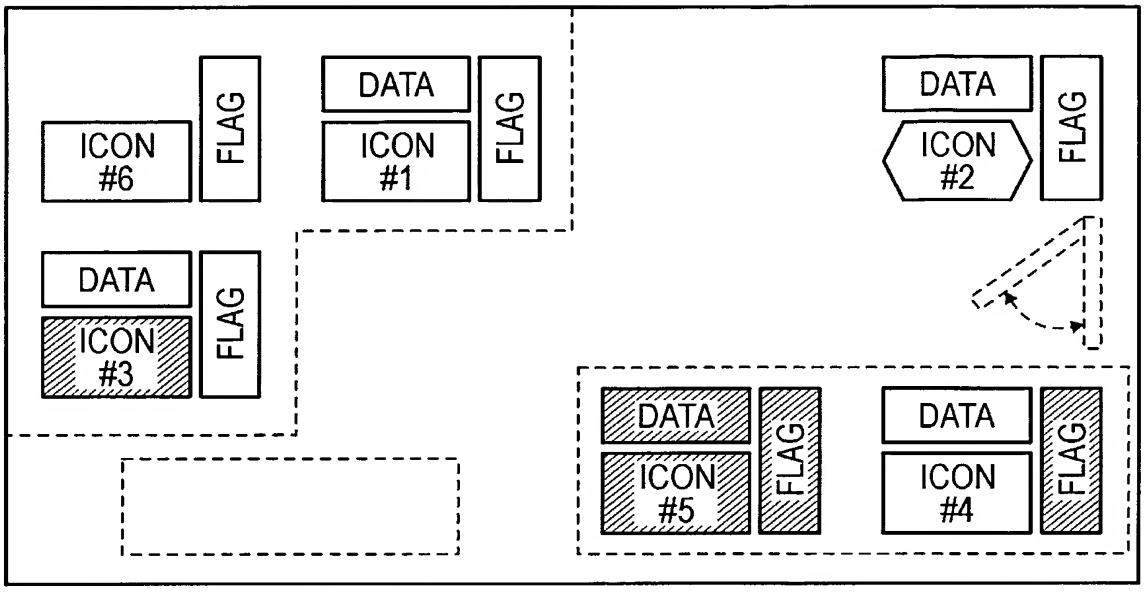

ICON #6
FLAG
DATA
ICON #1
FLAG
DATA
ICON #2
FLAG
DATA
ICON #3
FLAG
DATA
ICON #5
FLAG
DATA
ICON #4
FLAG

FIG. 8C

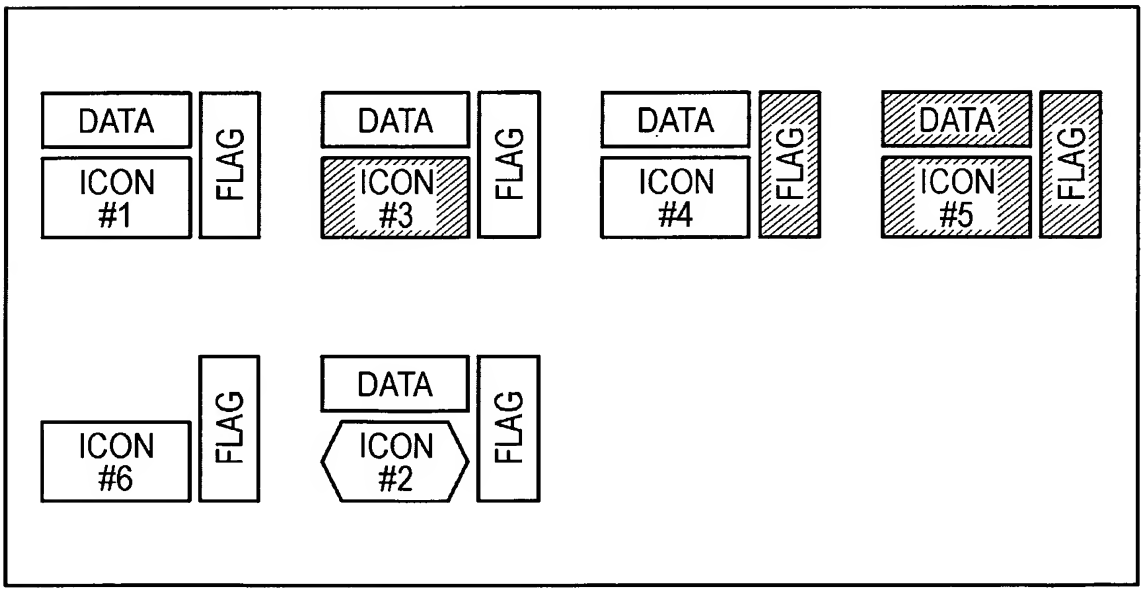

DATA
ICON #1
FLAG
DATA
ICON #3
FLAG
DATA
ICON #4
FLAG
DATA
ICON #5
FLAG
ICON #6
FLAG
DATA
ICON #2
FLAG

FIG. 8D

	DATA	PARAMETER #1	PARAMETER #2	PARAMETER #3	PARAMETER #4
NETWORK APPLIANCE #1	A	AA	BA	CA	DA
NETWORK APPLIANCE #1	A	AB	BB	CB	DB
NETWORK APPLIANCE #1	B	AC	BC	CC	DC
NETWORK APPLIANCE #1	B	AD	BD	CD	DD
NETWORK APPLIANCE #1	C	AE	BE	CE	DE

FIG. 9A

	DATA	PARAMETER #1	PARAMETER #2	PARAMETER #3	PARAMETER #4
NETWORK APPLIANCE #1	A	AA	BA	EA	DA
NETWORK APPLIANCE #1	A	AA	BB	EA	DB
NETWORK APPLIANCE #1	B	AA	BB	EA	DC
NETWORK APPLIANCE #1	B	AA	BD	CD	DD
NETWORK APPLIANCE #1	D	AA	BE	CE	DE

FIG. 9B

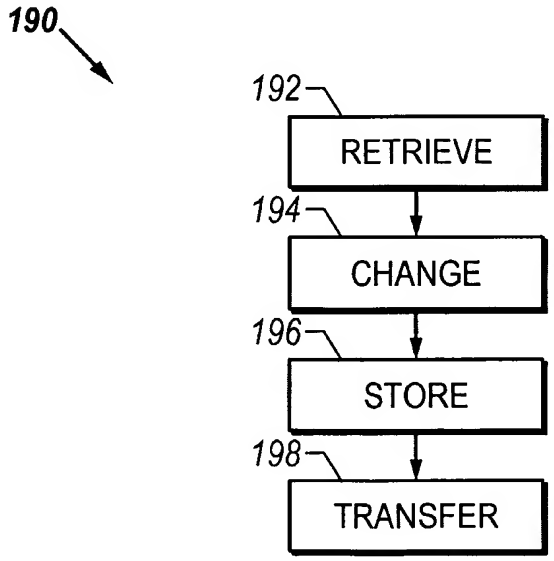


FIG. 10

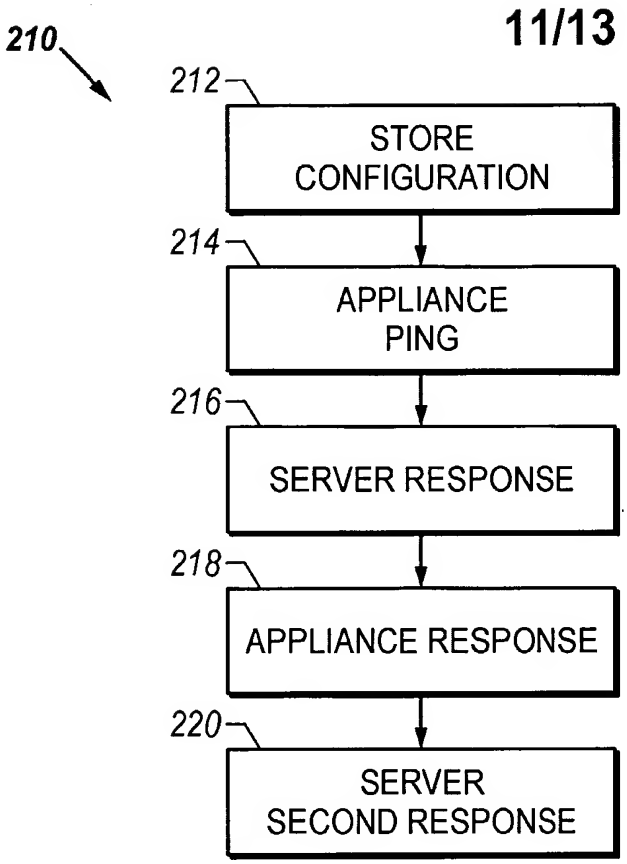

210
212
STORE CONFIGURATION
214
APPLIANCE PING
216
SERVER RESPONSE
218
APPLIANCE RESPONSE
220
SERVER SECOND RESPONSE

FIG. 11

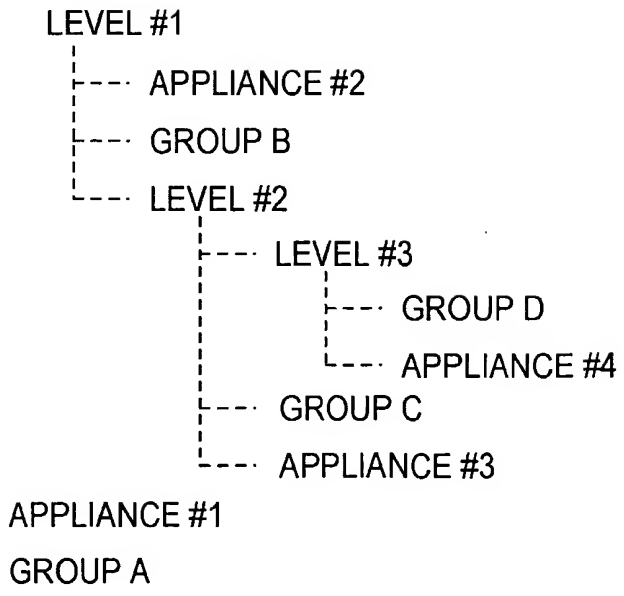

LEVEL #1
APPLIANCE #2
GROUP B
LEVEL #2
LEVEL #3
GROUP D
APPLIANCE #4
GROUP C
APPLIANCE #3
APPLIANCE #1
GROUP A

FIG. 12

	LOCATION	VALUE #1	VALUE #2	VALUE #3	STATUS
NETWORK APPLIANCE #1	A	AA	BA	CA	OFF
NETWORK APPLIANCE #1	A	AB	BB	CB	ON
NETWORK APPLIANCE #1	B	AC	BC	CC	ON
NETWORK APPLIANCE #1	B	AD	BD	CD	ON
NETWORK APPLIANCE #1	C	AE	BE	CE	ON

FIG. 13A

	LOCATION	VALUE #1	VALUE #2	VALUE #3	STATUS
NETWORK APPLIANCE #1	A	AA	BA	CA	ON
NETWORK APPLIANCE #1	A	AD	BB	CB	ON
NETWORK APPLIANCE #1	B	AC	BC	CC	ON
NETWORK APPLIANCE #1	B	AD	BD	CF	ON
NETWORK APPLIANCE #1	C	AE	BE	CE	OFF

FIG. 13B

	LOCATION	MODEL	VALUE #1	VALUE #2	STATUS
NETWORK APPLIANCE #1	A	100	AA	BA	OFF
NETWORK APPLIANCE #1	A	100	AB	BB	ON
NETWORK APPLIANCE #1	B	300	AC	BC	ON
NETWORK APPLIANCE #1	B	400	AD	BD	ON
NETWORK APPLIANCE #1	C	200	AE	BE	ON

FIG. 13C

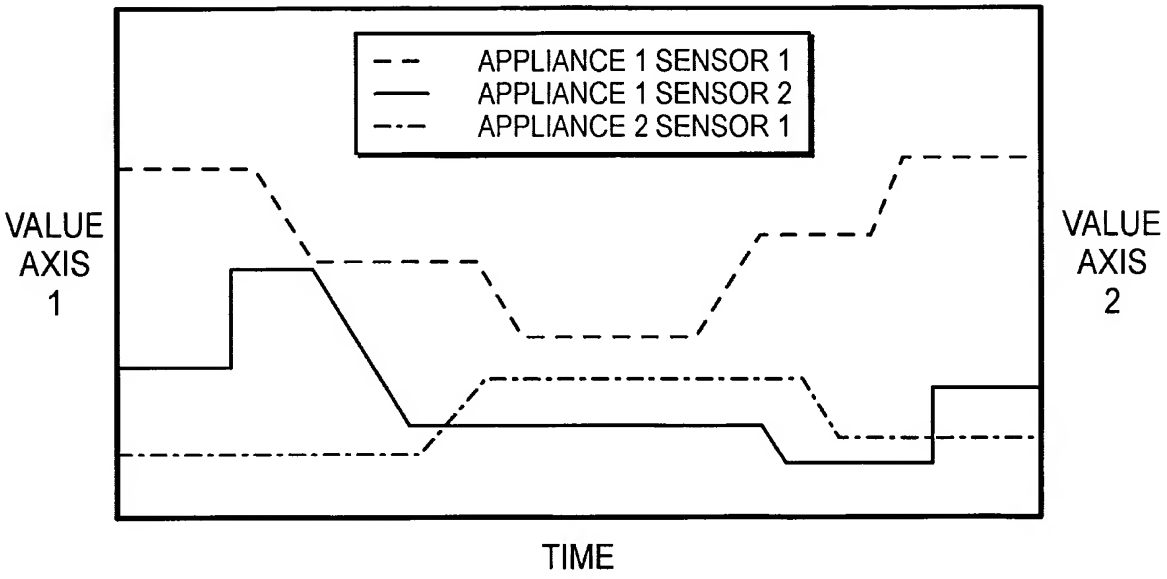

APPLIANCE 1 SENSOR 1
APPLIANCE 1 SENSOR 2
APPLIANCE 2 SENSOR 1
VALUE AXIS 1
VALUE AXIS 2
TIME

FIG. 14

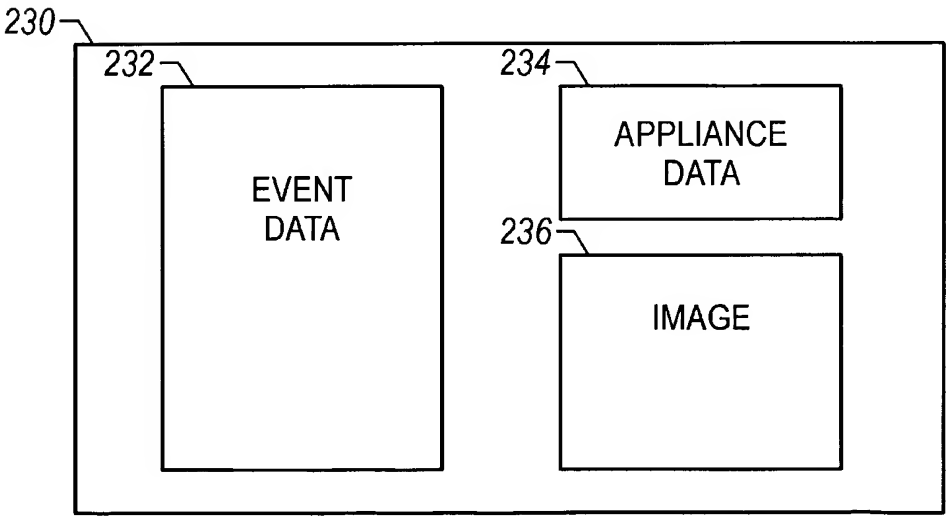

230
232
EVENT DATA
234
APPLIANCE DATA
236
IMAGE

FIG. 15

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US02/09179

A. CLASSIFICATION OF SUBJECT MATTER
IPC(7) : G06F 15/177
US CL : 709/221
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
U.S. : 709/221, 203, 220, 218

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Please See Continuation Sheet

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 6,167,406 A (HOSKINS et al.) 26 December 2000 (26.12.2000), col. 7, line 21 - col. 13, line 34, col. 24, line 10 - col. 26, line 4, col. 32, line 30 - col. 39, line 48, and col. col. 51, line 53 - col. 54, line 64.	1-37
A	US 5,097,328 A (BOYETTE) 17 March 1992 (17.3.1992), see entire document.	1, 8, 21, 30
A	US 5,220,522 A (WILSON et al.) 15 June 1993 (15.6.1993), col. 2, lines 20-49.	1, 8, 21, 30
A	US 5,659,470 A (GOSKA et al.) 19 August 1997, col. 4, line 40 - col. 5, line 53 and col. 11, line 49 - col. 12, line 64.	1, 8, 21, 30
A,P	US 6,259,956 B1 (MYERS et al.) 10 July 2001 (10.7.2001), col. 3, line 19 - col. 6, line 6.	1, 8, 21, 30

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
25 July 2002 (25.07.2002)

Date of mailing of the international search report
22 AUG 2002

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703)305-3230

Authorized officer
Jason Cardone
Peggy Harrod
Telephone No. (703) 305-3900

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT

PCT/US02/09179

Continuation of B. FIELDS SEARCHED Item 3:
STN
search terms: display, manipulate, client, server, sensor, java, applet